FOREWORD

Thank you very much for purchasing the CASIO FX-850P Personal Computer. This manual introduces and explains the scientific calculation function and BASIC programming language used with this computer. It is suggested that everyone from BASIC novices to veterans become familiar with the name and function of each part of the computer before attempting operation. Even when BASIC programming is not employed, a Formula Storage Function provides simplified formula calculations and ratio calculations, a built-in Data Bank Function allows memo handling and searches. Besides this, this computer also features a Built-in Scientific Utility which provides a total of 116 software utilities for statistical, mathematical and scientific applications.
PRECAUTIONS

This computer is a product of CASIO's high level of electronics engineering, testing, and quality control. The following points should be carefully noted to allow this unit to provide the years of trouble free operation for which it is designed.

- This unit is constructed of precision electronic components and should never be disassembled, dropped, or otherwise subjected to strong impact. Strong shocks can cause termination of program execution or alteration of the unit's memory.
- Do not use or store this unit in areas subjected to high temperatures, humidity or dust.
- Display response may become slow or fail completely at extremely low temperatures. Normal operation should resume after the unit reaches normal temperature.
- The connectors of this unit are designed exclusively for connection of the specified FA-6 expansion units only.
- The display may become dim when the buzzer sounds, but this does not indicate malfunction and is no cause for worry.
- Batteries should be replaced as soon as possible after weakened batteries are indicated by a dim display during normal operation.
- Replace batteries at least once every two years even if the unit is not used during this period. Dead batteries left in the unit may cause serious damage due to fluid leakage and should be removed as soon as possible.
- Keep the connector of the unit covered with the connector cap whenever the unit is not connected to an expansion unit, and avoid touching the connector.
- Strong static electrical charge may cause alteration of memory contents or key operation failure. If this situation should occur, remove the batteries and load them again.
- Always ensure that the power supply of this unit is switched OFF before connecting peripheral devices.
- Never use thinner, benzine, or other volatile agents for cleaning the exterior of the unit. Use a soft cloth dipped into a mild solution of water and a neutral detergent, and wring the cloth out completely.
- Do not switch the power of the unit OFF during program execution or during calculations.
- When a malfunction occurs, contact the store where the computer was purchased or a nearby dealer.
- If you are seeking service, please read this manual again, check the power supply, check the program for logic errors, etc.
# Contents

## Part 1 Unit Configuration

1-1 General Guide ................................................. 1
1-2 Operational Functions ................................. 2
1-3 Symbol Display ............................................. 4
1-4 Keyboard ...................................................... 5
  1-4-1 Keypad Functions .................................. 5
  1-4-2 Functions Noted Above the keys ................ 6
1-5 Screen ......................................................... 6
  1-5-1 Physical Lines and Logical Lines .............. 7
  1-5-2 Virtual Screen ...................................... 7
  1-5-3 Screen Editor ....................................... 7
  1-5-4 Display Contrast ................................... 8
1-6 Display Characters ........................................ 8
1-7 Power Supply ............................................... 9
1-8 Auto Power Off ........................................... 10
1-9 Connector .................................................... 10

## Part 2 Fundamental Operation

2-1 Cal Mode ..................................................... 11
2-2 Basic Mode .................................................. 11
2-3 Formula Storage Function ............................ 12
2-4 Data Bank Function (Memo in Mode) ............... 13
2-5 Built-in Scientific Library ......................... 14
2-6 Summary ....................................................... 14

## Part 3 Calculation Function

3-1 Manual Calculation Preparations ..................... 15
3-2 Manual Calculation Input and Correction ........... 15
3-3 Priority Sequence ....................................... 19
3-4 Scientific Calculations ................................ 20
3-5 Calculations Using Variables ....................... 29
3-6 Other Calculations ..................................... 30

## Part 4 Formula Storage Function

4-1 Utilization for Preparing Tables .................... 34
### Part 8: PB-100 Series Compatibility

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-1</td>
<td>PB-100 Series Program Input/Editing</td>
<td>77</td>
</tr>
<tr>
<td>8-2</td>
<td>PB-100 Series Program Execution</td>
<td>78</td>
</tr>
<tr>
<td>8-2-1</td>
<td>DEFM Mode</td>
<td>78</td>
</tr>
<tr>
<td>8-2-2</td>
<td>Using DEFM Statement Arrays</td>
<td>79</td>
</tr>
<tr>
<td>8-2-3</td>
<td>DEFM Mode Displays</td>
<td>80</td>
</tr>
<tr>
<td>8-2-4</td>
<td>CLEAR Command, DIM Command, DEFM Command</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>In DIM Mode and DEFM Mode</td>
<td></td>
</tr>
<tr>
<td>8-3</td>
<td>Loading PB-100 Series Programs</td>
<td>80</td>
</tr>
<tr>
<td>8-4</td>
<td>Reading PB-100 Series Data</td>
<td>81</td>
</tr>
<tr>
<td>8-5</td>
<td>Command Format</td>
<td>81</td>
</tr>
</tbody>
</table>

### Part 9: File Handling Fundamentals

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9-1</td>
<td>Filing Devices</td>
<td>83</td>
</tr>
<tr>
<td>9-2</td>
<td>File Descriptors</td>
<td>83</td>
</tr>
<tr>
<td>9-2-1</td>
<td>Cassette Tapes</td>
<td>83</td>
</tr>
<tr>
<td>9-2-2</td>
<td>Communications Circuit</td>
<td>84</td>
</tr>
</tbody>
</table>

### Part 10: Command Reference

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Format Elements</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>Manual Commands</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>Fundamental Commands</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>Numeric Functions</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>Character Functions</td>
<td>146</td>
</tr>
<tr>
<td></td>
<td>I/O Commands</td>
<td>158</td>
</tr>
<tr>
<td></td>
<td>Data Bank Commands</td>
<td>170</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>11-1</td>
<td>LIBRARY EXECUTION</td>
<td>176</td>
</tr>
<tr>
<td>11-1-1</td>
<td>Activating The Library</td>
<td>176</td>
</tr>
<tr>
<td>11-1-2</td>
<td>Library Termination</td>
<td>177</td>
</tr>
<tr>
<td>11-1-3</td>
<td>Library Activation Display</td>
<td>178</td>
</tr>
<tr>
<td>11-1-4</td>
<td>Examples Used in This Manual</td>
<td>179</td>
</tr>
<tr>
<td>11-1-5</td>
<td>Precautions When Using the Library</td>
<td>179</td>
</tr>
<tr>
<td>1000</td>
<td>MEMORY CALCULATIONS</td>
<td>180</td>
</tr>
<tr>
<td>5010</td>
<td>PRIME FACTOR ANALYSIS</td>
<td>183</td>
</tr>
<tr>
<td>5020</td>
<td>GREATEST COMMON MEASURE/LEAST COMMON MULTIPLE</td>
<td>184</td>
</tr>
<tr>
<td>5040</td>
<td>SIMULTANEOUS EQUATIONS (GAUSS-JORDAN ELIMINATION)</td>
<td>184</td>
</tr>
<tr>
<td>5050</td>
<td>QUADRATIC EQUATION</td>
<td>187</td>
</tr>
<tr>
<td>5060</td>
<td>CUBIC EQUATIONS</td>
<td>188</td>
</tr>
<tr>
<td>5080</td>
<td>NUMERIC SOLUTION OF AN EQUATION (NEWTON'S METHOD)</td>
<td>190</td>
</tr>
<tr>
<td>5090</td>
<td>NUMERIC SOLUTION OF AN EQUATION (BISECTION METHOD)</td>
<td>192</td>
</tr>
<tr>
<td>5100</td>
<td>MATRIX OPERATIONS</td>
<td>194</td>
</tr>
<tr>
<td>5200</td>
<td>NUMERIC INTEGRATION (ROMBERG'S METHOD)</td>
<td>204</td>
</tr>
<tr>
<td>5220</td>
<td>ORDINARY DIFFERENTIAL EQUATION (RUNGE-KUTTA METHOD)</td>
<td>206</td>
</tr>
<tr>
<td>5230</td>
<td>LAGRANGE'S INTERPOLATION</td>
<td>207</td>
</tr>
<tr>
<td>5250</td>
<td>GAMMA FUNCTION Γ(x)</td>
<td>208</td>
</tr>
<tr>
<td>5260</td>
<td>BESSEL FUNCTION Jn(x)</td>
<td>209</td>
</tr>
<tr>
<td>5270</td>
<td>BESSEL FUNCTION Yn(x)</td>
<td>210</td>
</tr>
<tr>
<td>5280</td>
<td>MODIFIED BESSEL FUNCTION In(x)</td>
<td>211</td>
</tr>
<tr>
<td>5290</td>
<td>MODIFIED BESSEL FUNCTION Kn(x)</td>
<td>212</td>
</tr>
<tr>
<td>5300</td>
<td>COMPLEX NUMBER</td>
<td>213</td>
</tr>
<tr>
<td>5350</td>
<td>BINARY-DECIMAL-HEXADECIMAL</td>
<td>217</td>
</tr>
<tr>
<td>5510</td>
<td>STRAIGHT LINE PASSING THROUGH TWO POINTS</td>
<td>222</td>
</tr>
<tr>
<td>5520</td>
<td>ANGLE OF INTERSECTION FOR TWO STRAIGHT LINES</td>
<td>223</td>
</tr>
<tr>
<td>5530</td>
<td>DISTANCE BETWEEN POINT AND STRAIGHT LINE</td>
<td>224</td>
</tr>
<tr>
<td>5540</td>
<td>ROTATIONAL MOVEMENT</td>
<td>225</td>
</tr>
<tr>
<td>5550</td>
<td>CIRCLE PASSING THROUGH THREE POINTS</td>
<td>226</td>
</tr>
<tr>
<td>5560</td>
<td>LENGTH OF TANGENT LINES FROM A POINT TO A CIRCLE</td>
<td>227</td>
</tr>
<tr>
<td>5570</td>
<td>TANGENT LINE EQUATION</td>
<td>228</td>
</tr>
<tr>
<td>5600</td>
<td>AREA OF A TRIANGLE</td>
<td>229</td>
</tr>
<tr>
<td>5605</td>
<td>AREA OF A TRAPEZOID</td>
<td>230</td>
</tr>
<tr>
<td>5610</td>
<td>AREA OF A PARALLELOGRAM</td>
<td>231</td>
</tr>
<tr>
<td>5615</td>
<td>AREA OF A CIRCLE</td>
<td>232</td>
</tr>
<tr>
<td>5620</td>
<td>AREA OF A SECTOR</td>
<td>233</td>
</tr>
<tr>
<td>5625</td>
<td>AREA OF A SEGMENT</td>
<td>234</td>
</tr>
<tr>
<td>5630</td>
<td>AREA OF AN ELLIPSE</td>
<td>235</td>
</tr>
</tbody>
</table>
6320  UPPER CUMULATIVE FREQUENCY (POISSON DISTRIBUTION) ........ 268
6330  UPPER CUMULATIVE FREQUENCY
       (HYPERGEOMETRIC DISTRIBUTION) .................................. 269
6410  PERCENTAGE POINT (NORMAL DISTRIBUTION) ......................... 290
6420  PERCENTAGE POINT (x² DISTRIBUTION) ................................ 291
6430  PERCENTAGE POINT (T DISTRIBUTION) ................................ 292
6440  PERCENTAGE POINT (F DISTRIBUTION) ................................ 293
6450  NORMAL RANDOM NUMBERS ............................................ 294
6460  EXPONENTIAL RANDOM NUMBERS ..................................... 295
6500  SINGLE VARIABLE STATISTICS .......................................... 296
6510  LINEAR REGRESSION ANALYSIS (y = a + bx) ............................ 299
6520  LOGARITHMIC REGRESSION ANALYSIS (y = a + blnx) ................. 302
6530  EXPONENTIAL REGRESSION ANALYSIS (y = ab^x) ....................... 305
6540  POWER REGRESSION ANALYSIS (y = ax^b) .............................. 308
6610  MEAN INTERVAL ESTIMATION (FOR KNOWN VARIANCE) ............... 312
6620  MEAN INTERVAL ESTIMATION (FOR UNKNOWN VARIANCE) .............. 315
6630  VARIANCE INTERVAL ESTIMATION ...................................... 318
6640  STANDARD DEVIATION INTERVAL ESTIMATION .......................... 321
6650  VARIANCE RATIO INTERVAL ESTIMATION ................................ 324
6660  MEAN DIFFERENCE INTERVAL ESTIMATION ............................... 328
6670  RATIO INTERVAL ESTIMATION .......................................... 332
6680  RATIO DIFFERENCE INTERVAL ESTIMATION ............................. 333
6710  POPULATION MEAN TEST (TWO-SIDED):
       FOR KNOWN VARIANCE .............................................. 335
6711  POPULATION MEAN TEST (RIGHT SIDED):
       FOR KNOWN VARIANCE .............................................. 338
6712  POPULATION MEAN TEST (LEFT SIDED):
       FOR KNOWN VARIANCE .............................................. 341
6720  POPULATION MEAN TEST (TWO-SIDED):
       FOR UNKNOWN VARIANCE ............................................ 344
6721  POPULATION MEAN TEST (RIGHT SIDED):
       FOR UNKNOWN VARIANCE ............................................ 347
6722  POPULATION MEAN TEST (LEFT SIDED):
       FOR UNKNOWN VARIANCE ............................................ 350
6730  POPULATION VARIANCE TEST (TWO-SIDED) ............................ 353
6731  POPULATION VARIANCE TEST (RIGHT SIDED) .......................... 356
6732  POPULATION VARIANCE TEST (LEFT SIDED) ........................... 359
6740  VARIANCE RATIO TEST (TWO-SIDED) .................................. 362
6741  VARIANCE RATIO TEST (RIGHT SIDED) ................................ 366
6742  VARIANCE RATIO TEST (LEFT SIDED) ................................ 370
6750  MEAN DIFFERENCE TEST (TWO-SIDED) ................................ 374
6751  MEAN DIFFERENCE TEST (RIGHT SIDED) ................................ 378
PART 12 APPENDICES ............................................. 395

12-1 CHARACTER CODE TABLE .................................. 395
12-2 ERROR MESSAGE TABLE .................................... 397
12-3 COMMAND/FUNCTION TABLE .............................. 399
12-4 RESERVED WORD LIST .................................... 400

SPECIFICATIONS ................................................. 401

BASIC COMMAND INDEX ......................................... 403
PART 1
UNIT CONFIGURATION

1-1 GENERAL GUIDE

- Power Switch
- Shift Key
- Numeric Keys
- Decimal Key
- Arithmetic Operator Keys
- Execute Key
- Parentheses Keys
- Answer Key
- Engineering Key

- Alphabet Keys
- Space Key
- CAPS Key
- Cursor Keys
- Insert/Delete Key
- Break Key
- Backspace/Clear Screen Key
- Menu Key
- LIB Key

- Memo Key
- Mode Key
- Program Area Keys
- Function Keys
- Formula Storage Key
- ALL RESET Button
- P Button
- Screen
1-2 OPERATIONAL FUNCTIONS

1. Power Switch (—□—□)
   Slides to the right to switch power ON and to the left to switch power OFF.

2. Shift Key (□)
   Used to enter BASIC commands and symbols noted in red on the key panel. Each press of this key causes the symbol "□" to switch ON and OFF on the display.
   * Throughout this manual, this key is represented by □ in order to distinguish it from the alphabetic □ key.

3. Numeric Keys (0 — 9)
   Enter the numeric values noted on each key.

4. Decimal Key (□)
   Enters a decimal point.

5. Arithmetic Operator Keys (+, —, ×, ÷)
   Enter the arithmetic operators noted on the keys.
   • +: Addition
   • —: Subtraction
   • ×: Multiplication
   • ÷: Division

6. Execute Key (□)
   Finalizes entry of a calculation and produces the result. The function of this key is equivalent to a "=" key on a standard calculator.
   This key is also used to enter lines of a program and for actual execution of programs.

7. Parentheses Keys (□□)
   Enter parentheses in such parenthetical calculations as: 5 × (10 + 20).

8. Answer Key (□□)
   Recalls the result of the most recently performed manual or program calculation. Pressing this key during program execution causes the execution to be suspended until the □□ key is pressed (STOP displayed).

9. Engineering Key (□□ / □□)
   Converts a calculation result to an exponential display.

10. Alphabet Keys
    Enter the alphabetic characters noted on each key.

11. Space Key (□□)
    Enters a space.

12. CAPS Key (□□)
    Switches the alphabet keys between upper case and lower case characters. The upper case mode is indicated by the "CAPS" symbol on the display.

13. Cursor Keys (□□ , □□ , □□ , □□)
    Move the cursor on the screen. Each press moves the cursor in the direction noted on the keys pressed, while holding down the keys causes continuous, high speed movement. Each cursor key also takes on a different function when pressed in combination with the □□ key.

<table>
<thead>
<tr>
<th>KEY</th>
<th>FUNCTION</th>
<th>□□ +</th>
</tr>
</thead>
<tbody>
<tr>
<td>□□</td>
<td>Cursor left</td>
<td>Moves to beginning of logical line</td>
</tr>
<tr>
<td>□□</td>
<td>Cursor right</td>
<td>Moves to end of logical line</td>
</tr>
<tr>
<td>□□</td>
<td>Cursor up</td>
<td>Scrolls screen up without cursor movement</td>
</tr>
<tr>
<td>□□</td>
<td>Cursor down</td>
<td>Scrolls screen down without cursor movement</td>
</tr>
</tbody>
</table>
4. Insert/Delete Key ( [ ] / [ ] )
Inserts a space at the current cursor position by shifting everything from the cursor position right one space to the right. In combination with the [ ] key, deletes the character at the current cursor position and automatically fills in the space created by shifting everything to the right of the cursor one space to the left. Holding down this key for either function causes continuous high speed operation of the respective function.

5. Break Key ( [ ] )
Terminates manual operations, program execution, printer output, and LIST output. Also reactivates the power supply when it has been interrupted by the Auto Power OFF function (see page 10).

6. Backspace/Clear Screen Key ( [ ] / [ ] )
Deletes the character located immediately to the left of the cursor and automatically fills in the space created by shifting everything from the cursor position right one space to the left. In combination with the [ ] key, clears the contents of the screen and locates the cursor at the upper left corner of the screen.

7. Menu Key ( [ ] )
Only operable in the CAL mode, this key is used to display a menu of built-in scientific library. See PART 11 SCIENTIFIC LIBRARY for details.

8. LIB Key ( [ ] )
Only operable in the CAL mode, this key executes the operation corresponding to an entered library number. See PART 11 SCIENTIFIC LIBRARY for details.

9. Memo Key ( [ ] )
Used to display and search for DATA BANK data. See PART 5 DATA BANK FUNCTION for details.

10. Mode Key ( [ ] )
Used in combination with numeric keys to specify operational modes.

- [ ] ......... CAL mode (selected when power is switched ON)
- [ ] ......... BASIC mode (program writing/editing)
- [ ] ......... DEG mode (angle unit = degrees)
- [ ] ......... RAD mode (angle unit = radians)
- [ ] ......... GRA mode (angle unit = grads)
- [ ] ......... Print ON
- [ ] ......... Print OFF
- [ ] ......... MEMO IN mode (DATA BANK function)

11. Program Area Keys ( [ ] - [ ] )
Executes the program in the corresponding program area in the CAL mode. Specifies a program area for writing or editing in the BASIC mode.

12. Function Keys ( [ ] , [ ] , [ ] , etc.)
Allow one-touch entry of often used functions.

- Direct input functions
  [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ]
- Enhanced functions
  [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ]

13. Formula Storage Keys ( [ ] , [ ] , [ ] )
Used when working with the formula storage function. See PART 4 FORMULA STORAGE FUNCTION for details.

14. ALL RESET Button ( [ ] )
Clears all memory contents and enters the CAL mode. All important data should be saved elsewhere before pressing this button. If pressing this button does not clear memory contents, first press the P button and then press the ALL RESET button again.
P Button ( ) (rear panel)
Hardware reset button to halt misoperation caused by static electricity. Though execution
is interrupted, memory contents are retained. The ALL RESET button should be used when
the misoperation damages memory contents. Note that power switches OFF and then ON
again when the P button is pressed.

Screen
A 32-column x 2-line liquid crystal display upon which 5 x 7-dot characters appear.

1-3 SYMBOL DISPLAY

The symbols noted on the display illustrated below appear to show the current status of a
calculation.

Memo data record number (MEMO mode)

| CAPS & CAL BASIC DECIMALIAN NUM IN ED TR DEF SA SAT TO STOP |

CAPS : Upper case alphabetic characters (lower case when not displayed)
5 : Shift mode (commands/functions marked above keys can be input)
CAL : CAL mode (basic arithmetic calculations or function calculations)
BASIC : BASIC mode (BASIC program input, editing, execution)
DEG : Angle unit = degrees
RAD : Angle unit = radians
GRA : Angle unit = grads
MEMO : DATA BANK data search, display, input, editing
IN : DATA BANK data input, editing
EDIT : DATA BANK data editing, BASIC program editing (using EDIT command)
DEFM : DEFM mode (for execution of CASIO PB-100 series programs)
LIB : Scientific library function mode
PRT : Print mode (output of display contents to printer)
TR : Trace mode (traces execution of BASIC programs)
STOP : Display suspended (by BASIC STOP command or PRINT statement)
A look at the keyboard of the unit reveals characters and symbols located above the keys. These are accessed using the \textcolor{red}{\text{PA}} and \textcolor{green}{\text{PG}} keys.

1-4-1 Keytop Functions

Normal Mode

In this mode, each key inputs the characters, symbols, or commands noted on the keys themselves. (This status is automatically set when power is switched ON and immediately following the ALL RESET procedure.)

EXAMPLE:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>E</td>
<td>E</td>
</tr>
</tbody>
</table>
Lower Case Mode

Pressing the A key shifts the alphabetic keys (only) to lower case characters, indicated by the CAPS symbol disappearing from the display. Pressing the A key once locks the keyboard into the lower case mode, while pressing again returns to upper case.

EXAMPLE:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Q w e r t y u i o p</td>
</tr>
<tr>
<td>b</td>
<td>a s d f g h j k l</td>
</tr>
<tr>
<td>d</td>
<td>z x c v b n m</td>
</tr>
</tbody>
</table>

1-4-2 Functions Noted Above the Keys

The BASIC one-key commands, and the symbols and commands noted above the keys are entered when the corresponding keys are pressed following the A key. Note, however, that pressing the numeric keys (0 - 5) after A in the CAL mode executes the program in the corresponding program area, while, in the BASIC mode, switches to the corresponding program area.

EXAMPLE:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>w / n f a c t</td>
</tr>
<tr>
<td>b</td>
<td>p m s d</td>
</tr>
<tr>
<td>c</td>
<td>p m s d</td>
</tr>
<tr>
<td>d</td>
<td>p m s d</td>
</tr>
<tr>
<td>e</td>
<td>p m s d</td>
</tr>
<tr>
<td>f</td>
<td>p m s d</td>
</tr>
<tr>
<td>g</td>
<td>p m s d</td>
</tr>
<tr>
<td>h</td>
<td>p m s d</td>
</tr>
<tr>
<td>i</td>
<td>p m s d</td>
</tr>
<tr>
<td>j</td>
<td>p m s d</td>
</tr>
<tr>
<td>k</td>
<td>p m s d</td>
</tr>
<tr>
<td>l</td>
<td>p m s d</td>
</tr>
<tr>
<td>m</td>
<td>p m s d</td>
</tr>
<tr>
<td>n</td>
<td>p m s d</td>
</tr>
<tr>
<td>o</td>
<td>p m s d</td>
</tr>
<tr>
<td>p</td>
<td>p m s d</td>
</tr>
<tr>
<td>q</td>
<td>p m s d</td>
</tr>
<tr>
<td>r</td>
<td>p m s d</td>
</tr>
<tr>
<td>s</td>
<td>p m s d</td>
</tr>
<tr>
<td>t</td>
<td>p m s d</td>
</tr>
<tr>
<td>u</td>
<td>p m s d</td>
</tr>
<tr>
<td>v</td>
<td>p m s d</td>
</tr>
<tr>
<td>w</td>
<td>p m s d</td>
</tr>
<tr>
<td>x</td>
<td>p m s d</td>
</tr>
<tr>
<td>y</td>
<td>p m s d</td>
</tr>
<tr>
<td>z</td>
<td>p m s d</td>
</tr>
</tbody>
</table>

1-5 SCREEN

The screen is a 32-column x 2-line liquid crystal display. Characters are formed by a 5 x 7 dot matrix.
1-5-1 Physical Lines and Logical Lines

The maximum display capacity of one line is 32 columns, but internally the unit is capable of handling lines up to 255 characters long. The display capacity line (32 characters) is referred to as the physical line, while the internal capacity line is called a logical line. A logical line is a continuous line of characters in which any column on the extreme right of the screen is not a null.

![Physical and Logical Lines Diagram]

Pressing $\text{Esc}$ moves the cursor to the beginning of the logical line, while $\text{Shift}$ moves the cursor to the end of the logical line. These operations are useful in determining the extent of logical lines.

1-5-2 Virtual Screen

The screen can display two lines at one time, and as the 3rd line is input, the first line scrolls off the top of the screen. Lines that scroll off of the screen can, however, be brought back into view using the cursor ($\text{Up} / \text{Down}$) keys, because the unit is able to store up to eight lines internally. These eight lines make up the virtual screen, while the two lines actually displayed are called the actual screen.

![Virtual and Actual Screen Diagram]

1-5-3 Screen Editor

Any program lines or data included on the virtual screen can be edited. First the portion of the program or data is brought onto the actual screen, and then the cursor is located at the position to be edited.
1-5-4 Display Contrast

The display may appear dark or dim depending upon the strength of the batteries or the viewing angle. The contrast of the display can be adjusted to the desired level by rotating the control dial. Rotating the dial down (arrow direction) darkens the display, while rotating it up lightens the display.

A weak display when contrast is set to a high level indicates weakened batteries, and batteries should be replaced as soon as possible (see page 9).

1-6 DISPLAY CHARACTERS

The relationship between characters and character codes is illustrated in the following table.

Character Code Table

<table>
<thead>
<tr>
<th>High-order digit</th>
<th>0</th>
<th>16</th>
<th>32</th>
<th>48</th>
<th>64</th>
<th>80</th>
<th>96</th>
<th>112</th>
<th>128</th>
<th>144</th>
<th>160</th>
<th>176</th>
<th>192</th>
<th>208</th>
<th>224</th>
<th>240</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-order digit</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
<td>A</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>B</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt;</td>
</tr>
</tbody>
</table>

* Blank segments are not output.
* Notations in parentheses are control codes and are not displayed.
* Characters which cannot be displayed using keyboard input can be displayed using the CHRS function.
1-7 POWER SUPPLY

This unit is equipped with a main power supply (two CR2032 lithium batteries) and a back up power supply (one CR1220 lithium battery). Batteries should be replaced whenever the display remains dim, even after contrast adjustment. Batteries should also be replaced once every two years regardless of how much the unit has been used.

- Battery Replacement

1) Switch the power of the unit OFF and remove the rear panel of the unit after removing the two screws holding it in place.

2) Remove the main battery holder by removing screw A, and/or the back up battery holder by removing screw B.

3) Remove the old batteries by turning the unit so that the battery compartment faces downward and tapping gently.

4) Wipe the surfaces of new batteries with a dry cloth and load them into the battery compartment ensuring that the positive pole (+) is facing upwards.

5) Replace the battery holder screw while pressing down on the batteries.

6) Replace the rear panel of the unit and two screws to hold it in place.

* Simultaneously removing both the main battery and back up battery causes programs and data stored in memory to be deleted.
* Be sure to remove dead batteries from the unit to avoid damage due to battery leakage.
* Be sure to replace both main batteries.
* Always ensure that battery polarity is correct.
* Never dispose of batteries in such a way that they will be incinerated. Exposing batteries to high heat can cause them to explode.
* P Button
  Hardware reset button to halt misoperation caused by static electricity. Be sure to switch power OFF and ON when the P button is pressed.

* Keep batteries out of the reach of small children. Consult a physician immediately if inadvertently swallowed.
1-8 AUTO POWER OFF

The power of the unit is automatically switched OFF approximately 6 minutes after the last key operation (except during program execution), or the last input for an INPUT statement or PRINT statement. Power can be resumed by either switching the power switch OFF and then ON again, or by pressing the ON key.

* Program and data contents are retained even when power is switched OFF, but settings such as the number of digits or the mode (i.e. BASIC mode, MEMO IN mode) are canceled.

1-9 CONNECTOR

The connector noted in the illustration is equipped for connection of peripheral devices (i.e. FA-6). Be sure to leave the connector cover in place when peripheral devices are not connected.
This section covers the various modes available with the FX-850P using a series of simple examples. These procedures should be mastered before attempting more complex operations.

2-1 CAL MODE

The CAL mode is in effect each time the power of the unit is switched ON. Arithmetic calculations, function calculations, scientific library execution, formula storage calculations, program execution, and data recall can be performed in this mode.

EXAMPLE:

2.5 + 3.5 - 2 =

OPERATION:

2 \[ \boxed{+} \] 5 \[ \boxed{+} \] 3 \[ \boxed{-} \] 5 \[ \boxed{-} \] 2 \[ \boxed{\text{exit}} \]

Though the \[ \boxed{\text{exit}} \] key is used instead of the \[ \boxed{\text{AC}} \] key, operation is identical to that used in a standard calculator.

The CAL mode can be entered from another mode by pressing \[ \boxed{\text{CAL}} \]. See PART 3 CALCULATION FUNCTION (page 15) for details.

2-2 BASIC MODE

The BASIC mode is used for the creation, execution and editing of BASIC programs. The BASIC mode can be entered from another mode by pressing \[ \boxed{\text{BASIC}} \].

EXAMPLE:

Create and execute a program which calculates the sum of two values A and B.
PROGRAM INPUT

1

10 A = 5 \times

20 B = 6 \times

30 P R I N T A = B \times

40 E N D \times

PROGRAM EXECUTION

RUN

See PART 6 BASIC PROGRAMMING (page 45) for details on using the BASIC language.

2-3 FORMULA STORAGE FUNCTION

The formula storage function makes it possible to store often used formulas in memory for calculation when values are assigned to variables. This function is applied in the CAL mode using the \( \times \), \( \times \), and \( \times \) keys.

EXAMPLE:

Determine the selling price of a product by applying a profit rate based on the purchase price and selling price.

\[
\text{SELLING PRICE} = \text{PURCHASE PRICE} \div (1 - \text{PROFIT})
\]

KEY INPUT

\text{\textbullet} 1

\text{SELL LR CALC HAS E / (1 - PR O F I T)}

Required to store formula in memory

Ensure that input of the formula is correct by pressing the \( \text{\textbullet} \) key.

OPERATION:

\text{SELL LR CALC HAS E / (1 - PR O F I T) \text{\textbullet}}
Now calculate the selling prices of the following:

<table>
<thead>
<tr>
<th>PURCHASE PRICE</th>
<th>PROFIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1,000</td>
<td>30%</td>
</tr>
<tr>
<td>$960</td>
<td>25%</td>
</tr>
</tbody>
</table>

As can be seen in this example, once a formula is input, it can be used repeatedly by simply assigning values for the variables. See PART 4 FORMULA STORAGE FUNCTION (page 33) for details.

* The \text{mem} key can be used to terminate this function.

### 2-4 DATA BANK FUNCTION (MEMO IN MODE)

The DATA BANK function allows the storage of large volume of data for recall when required. Pressing \text{mem} \text{c} enters the MEMO IN mode, and causes the cursor to appear at the upper left of the display waiting for data input.

**EXAMPLE:**

J. SMITH 347-237-4811

\text{mem} \text{c}
\text{mem} \text{S}
\text{mem} \text{d} \text{M} \text{d} \text{m} \text{d} \text{j} \text{d} \text{m} \text{d} \text{S} \text{mem} \text{347} \text{mem} \text{2} \text{mem} \text{3} \text{mem} \text{7} \text{mem} \text{4} \text{mem} \text{8} \text{mem} \text{1} \text{mem} \text{1} \text{mem} \text{c}

Once data are stored, specific data items can be recalled using the \text{mem} key.

\text{mem} \text{0} \hspace{1cm} \text{(CAL mode)}

\text{mem} \text{S} \hspace{1cm} \text{SMITH J. 347-237-4811} \text{mem} \text{c} \hspace{1cm} \text{(Search for data item beginning with "S").}

The \text{mem} key can be used to terminate the DATA BANK function. For details on the storage and retrieval of data, see PART 5 DATA BANK FUNCTION (page 37).
2-5 BUILT-IN SCIENTIFIC LIBRARY

This function provides a wide variety of useful scientific library that can be recalled and used in calculations in the CAL mode.
Mathematical/Statistical operations — 116 types
Operations are selected using the A, B, C, D, and E keys.
* For details, see PART 11 SCIENTIFIC LIBRARY (page 176).

2-6 SUMMARY

Function Table

<table>
<thead>
<tr>
<th>Function name</th>
<th>Key operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL mode</td>
<td>E</td>
</tr>
<tr>
<td>BASIC mode</td>
<td>F</td>
</tr>
<tr>
<td>Data bank</td>
<td>G</td>
</tr>
<tr>
<td>Formula storage</td>
<td>C + A, B, C, D</td>
</tr>
<tr>
<td>Built-in scientific library</td>
<td>E + Library No. G</td>
</tr>
</tbody>
</table>
This section covers fundamental arithmetic calculations and function calculations which are performed manually.

3-1 MANUAL CALCULATION PREPARATIONS

Switch the Power of the Unit ON

The display illustrated above appears whenever the power is switched ON. It indicates the CAL mode in which manual calculations can be performed. Currently specified angle unit, however, is retained even when the power is switched OFF.

3-2 MANUAL CALCULATION INPUT AND CORRECTION

Perform the following fundamental calculations to become familiar with this mode.

EXAMPLE:

\[ 123 + 456 = 579 \]

\[ 123 + 456 \]

\[ 123 \times 456 \]

\[ 579 \]

As can be seen here, the \[ \text{M} \] key is pressed in place of \[ \text{D} \]. The \[ \text{K} \] key is used for multiplication and \[ \text{D} \] is used for division.

The following procedure can be used to correct entered data.

EXAMPLE:

\[ 33 \times 5 + 16 = 181 \]

For the sake of example, the value 33 here will be mistakenly entered as 34.
34 5 16

Press \( \rightarrow \) six times to move cursor back to position of 4. This can also be accomplished by \( \leftarrow \).

\[
\begin{array}{c}
34 \times 5 + 16 \\
33 \times 5 + 16 \\
33 \times 5 + 16 \\
\end{array}
\]

(Replaces 4 with 3.)

EXAMPLE:

\[
33 \times 5 + 16 = 181
\]

For the sake of example, the above calculation will be performed with the value 33 mistakenly entered as 34.

\[
\begin{array}{c}
34 \times 5 + 16 \\
34 \times 5 + 16 \\
\end{array}
\]

(Move cursor to position for correction.)

\[
\begin{array}{c}
3 \times 5 + 16 \\
33 \times 5 + 16 \\
\end{array}
\]

(Reexecute calculation.)

EXAMPLE:

\[
33 \times 5 + 16 = 181
\]

For the sake of example, the multiplication sign (*) here will be mistakenly omitted and calculated.

\[
\begin{array}{c}
335 + 16 \\
335 + 16 \\
\end{array}
\]

(Move cursor to position for insert of \( \times \).)

\[
\begin{array}{c}
335 + 16 \\
335 + 16 \\
\end{array}
\]

(Opens up one character space.)

\[
\begin{array}{c}
335 + 16 \\
335 + 16 \\
\end{array}
\]

(Enter multiplication symbol.)

\[
\begin{array}{c}
335 + 16 \\
335 + 16 \\
\end{array}
\]

(Obtain result again.)

As can be seen in the above example, the \( \rightarrow \) key is used to insert spaces at the current cursor location for input of characters or symbols.
EXAMPLE:

33 × 5 + 16 = 181

For the sake of example, the above calculation will be performed with the value 16 mistakenly entered as 216.

33 [×] 5 [+] 216 [EX]

33 × 5 + 216
381

(Moves cursor back to position of 2)

33 × 5 + 16
381

(Delete 2.)

33 × 5 + 16
181

(Reexecute calculation)

As can be seen in the above example, the [DEL] key is used to delete characters at the current cursor location.

• The [ES] key can also be used to delete characters, but its operation is slightly different from [DEL].

[DEL] ......... Deletes the character at the current cursor location.

ABCQEFGH

[DEL]

ABQEFGH

[ES] .............. Deletes the character at the position to the left of the current cursor location.

ABCQEFGH

[ES]

ABQEFGH

Practice the following examples to become familiar with the fundamental calculation procedure.

EXAMPLE 1:

9 + 7.8 ÷ 6 - 3.5 × 2 = 3.3

OPERATION:

9 [+] 7.8 [/] 6 [-] 3.5 [×] 2 [EX]

9 + 7.8 / 6 - 3.5 × 2
3 3

EXAMPLE 2:

56 × (−12) ÷ (−2.5) = 268.8
OPERATION:

\[
\begin{array}{c}
56 \div 12 \div 2.5 \text{ ENG} \\
\frac{56 \div 12}{2.5} = 22.8
\end{array}
\]

Negative values are entered by pressing the \(\text{ENG}\) key before entering the value.

EXAMPLE 3:

\[
(4.5 \times 10^{75}) \times (-2.3 \times 10^{-75}) = -0.01035
\]

OPERATION:

\[
\begin{array}{c}
4.5 \text{ ENG} 75 \div 2.3 \text{ ENG} 78 \text{ ENG} \\
4.5 \times 10^{75} \div 2.3 \times 10^{-78} = -0.01035
\end{array}
\]

Exponents are entered by pressing the \(\text{ENG}\) key (or the alphabetic \(\text{ENG}\) key) before entering the value.

The following example shows how the result of one calculation can be immediately incorporated into a subsequent calculation.

EXAMPLE 4:

\[
(23 + 456) \times 567 = 271593
\]

OPERATION:

\[
\begin{array}{c}
23 \text{ ENG} 456 \text{ ENG} \\
23 + 456 = 479
\end{array}
\]

continuing with

\[
\begin{array}{c}
479 \times 567 = 271593
\end{array}
\]

The last result obtained can be entered at any point in a subsequent calculation by pressing the \(\text{ENG}\) key.

EXAMPLE 5:

\[
81.2 \div (5.6 + 8.9) = 5.6
\]

\[\text{This process performed first}\]

OPERATION:

\[
\begin{array}{c}
5.6 \text{ ENG} 8.9 \text{ ENG} \\
5.6 + 8.9 = 14.5
\end{array}
\]

continuing with

\[
\begin{array}{c}
14.5 \div 81.2 = 0.17 \quad 81.2 \div 14.5 = 5.6
\end{array}
\]
3-3 PRIORITY SEQUENCE

Arithmetic, relational and logical operations are performed in the following priority sequence:
1. \((, )\)
2. Functions
3. Power
4. Signs \((+, -)\)
5. \(^, /, \%, \text{MOD}\)
6. \(+, -\)
7. Relational operators
8. \(\text{NOT}\)
9. \(\text{AND}\)
10. \(\text{OR}, \text{XOR}\)

EXAMPLE:

\[
2 + 4 \times \cos (14 + 16)^2 = 5
\]

NOTE:

a. Calculations are performed from left to right when the priority sequence is identical.
b. Complex functions (\(\sin \cos 60\)) are performed from right to left.
c. Consecutive powers \((5^4^3)\) is performed from left to right.

Calculation results are displayed in the following manner:
1. Integer less than \(1 \times 10^{10}\): Integer
2. 10 digits or less in fractional part: Decimal
3. Others: Exponential

Number of Digits

- Internal calculations are performed with a 12-digit mantissa + 2-digit exponent. PI, however, is expressed in 11 digits (3.1415926536). Rounded and displayed in 10 digits (3.141592654).
- Calculation results are displayed rounded off to a 10-digit mantissa + 2-digit exponent.
- Up to 255 characters can be entered for a single line.

Specifying the Number of Significant Digits and the Number of Decimal Places

"SET" is used for these specifications.

Specification of number of decimal places ............... \(\text{SET} \ (F) \ (0-9) \text{ DEC}\)
Specification of number of significant digits ............... \(\text{SET} \ (E) \ (0-9) \text{ DEC}\)

* "SET E0" used to specify the number of significant digits specifies 10 digits.

Release of specification ........................... \(\text{SET} \ (N) \text{ DEC}\)

* When a specification is made, the result is displayed by the number of specified digits.
(The digit next to the last specified digit is rounded off.) The original value remains in the computer.
EXAMPLE:

Specified number of decimal places: 2

OPERATION:

\[
\begin{array}{c}
\text{SET F2} \\
10 \div 3 \\
\end{array}
\]

\[
\begin{array}{c}
10 / 3 \\
3.33 \\
\end{array}
\]

EXAMPLE:

Specified number of significant digits: 3

OPERATION:

\[
\begin{array}{c}
\text{SET E3} \\
12 \times 34 \\
\end{array}
\]

\[
\begin{array}{c}
12 \times 34 \\
4.06E+02 \\
\end{array}
\]

3-4 SCIENTIFIC CALCULATIONS

The scientific functions (see the scientific function table on page 28) can be used either within programs or for manual calculations. For the sake of explanation, all of the examples here will cover only manual calculations.

Trigonometric and Inverse Trigonometric Functions

- \( \sin \) \: sine
- \( \cos \) \: cosine
- \( \tan \) \: tangent
- \( \sin^{-1} \) \: arc sine
- \( \cos^{-1} \) \: arc cosine
- \( \tan^{-1} \) \: arc tangent

These functions return a trigonometric function value for a given angle, or an angle value of a given trigonometric function value. The ANGLE command should be used to specify the unit for the angle value when these functions are used. Angle unit specification is only required once for all subsequent trigonometric/inverse trigonometric functions. Angle units can be specified using either the \( \equiv \) key or the ANGLE command.

- ANGLE 0......DEG (degrees) \( \rightarrow \) (4)
- ANGLE 1......RAD (radians) \( \rightarrow \) (5)
- ANGLE 2......GRAD (grads) \( \rightarrow \) (6)

The relationship among these three specifications is:

90 degrees = \( \frac{\pi}{2} \) radians = 100 grads

The current angle unit is retained when the power of the unit is switched OFF, and the angle unit becomes ANGLE 0 (DEG) when the ALL RESET button is pressed.

The value for \( \pi \) can be directly entered into a formula using "PI" (3.141592654).
EXAMPLE 1:

\[ \sin 30^\circ = 0.5 \]

**OPERATION:**

\[ \text{SIN} \]

\[ 30 \]

\[ = \]

\[ 0.5 \]

EXAMPLE 2:

\[ \cos \frac{\pi}{3} = 0.5 \]

**OPERATION:**

\[ \text{COS} \]

\[ \left( \frac{\pi}{3} \right) \]

\[ = \]

\[ 0.5 \]

**Note:** \( \pi \) can also be entered as \( \text{P} \ 1 \).

EXAMPLE 3:

\[ 2\sin \frac{\pi}{3} + \cos \frac{\pi}{3} = 2.232050808 \]

**OPERATION:**

\[ 2 \]

\[ \times \]

\[ \text{SIN} \]

\[ \left( \frac{\pi}{3} \right) \]

\[ + \]

\[ \text{COS} \]

\[ \left( \frac{\pi}{3} \right) \]

\[ = \]

\[ 2.232050808 \]
EXAMPLE 4:
\[ \tan 60^\circ = 1.732050808 \]

OPERATION:

\[ \begin{align*}
\text{[Enter] 4} \\
\text{[Enter] (or [T A N]) 60 [EXE]}
\end{align*} \]

\[ \text{[TAN]} \]
\[ \text{1.732050808} \]

EXAMPLE 5:
\[ \sin^{-1} 0.5 = 30^\circ \]

OPERATION:

\[ \begin{align*}
\text{[Enter] 0.5 [EXE]}
\end{align*} \]

\[ \text{[ASN]} \]
\[ 0.5 \]
\[ 30 \]

EXAMPLE 6:
\[ \cos^{-1} \frac{\sqrt{2}}{2} = 45^\circ \]

OPERATION:

\[ \begin{align*}
\text{[Enter] 1 [EXE] 2 [DIV] 0.5 [DIV] 2 [2] [EXE]}
\end{align*} \]

\[ \text{[ACS]} \]
\[ (2^0 \cdot 0.5/2) \]
\[ 45 \]

EXAMPLE 7:
\[ \tan^{-1} \sqrt{3} = 60^\circ = 1.047197551 \]

OPERATION:

\[ \begin{align*}
\text{[Enter] 3 [EXE]}
\end{align*} \]

\[ \text{[ATNSQR3]} \]
\[ 60 \]

\[ \text{[Enter] 5} \]

\[ \text{[ATNSQR3]} \]
\[ 60 \]

\[ \begin{align*}
\text{[Enter] 3 [EXE]}
\end{align*} \]

\[ \text{[ATNSQR3]} \]
\[ 1.047197551 \]
Hyperbolic and Inverse Hyperbolic Functions

\[
\begin{align*}
\sinh \, \text{hyperbolic sine} & \quad \sinh^{-1} \, \text{hyperbolic arc sine} \\
\cosh \, \text{hyperbolic cosine} & \quad \cosh^{-1} \, \text{hyperbolic arc cosine} \\
\tanh \, \text{hyperbolic tangent} & \quad \tanh^{-1} \, \text{hyperbolic arc tangent}
\end{align*}
\]

EXAMPLE 1:

\[\sinh 5 = 74.20321058\]

OPERATION:

\[\text{HYPSIN5} \quad 74.20321058\]  
(The HYPSIN function is used for \(\sinh\).)

EXAMPLE 2:

\[\cosh^{-1} 1.5 = 0.9624236501\]

OPERATION:

\[\text{HYPACS1} \quad 0.9624236501\]  
(The HYPACS function is used for \(\cosh^{-1}\).)

Logarithmic Functions, Exponential Functions

\[
\begin{align*}
\log_{10} \, & \text{common logarithm} \\
\log_{e} \, & \text{natural logarithm} \\
e^{x} \, & \text{exponent}
\end{align*}
\]

EXAMPLE 1:

\[\log_{e} 123 = 4.812184355\]

OPERATION:

\[\text{LN123} \quad 4.812184355\]  
(The LN function is used for \(\log_{e}\).)

EXAMPLE 2:

\[\log_{e} 100 = 2\]

OPERATION:

\[\text{LOG100} \quad 2\]  
(The LOG function is used for \(\log_{e}\).)

EXAMPLE 3:

\[e^{4} = 148.4131591\]

OPERATION:

\[\text{EXP5} \quad 148.4131591\]  
(The EXP function is used for \(e^{x}\).)
Other Functions

SGN: Sign \[ \text{SGN} \] Random number \[ \text{RAN} \] Absolute value \[ \text{ABS} \]
INT: Integer value \[ \text{INT} \] Integer part \[ \text{FIX} \]
ROUND: Rounding \[ \text{FRAC} \]

- **SGN**
  For SGN (x), returns a 1 when x > 0, a -1 when x < 0, and a 0 when x = 0.

  **OPERATION:**

  \[
  \begin{array}{c}
  \text{SGN6} \rightarrow 1 \\
  \text{SGN} - 2 \rightarrow -1
  \end{array}
  \]

- **RAN #**
  Generates a random number between 0 and 1 with up to 10 decimal places. For details, see PART 10 COMMAND REFERENCE.

  **OPERATION:**

  \[
  \begin{array}{c}
  \text{RAN} \cdot (\text{-1}) \rightarrow 0.2456393888 \times
  \end{array}
  \]
  * The above is only a sample value.

- **ABS**
  Returns the absolute value of x for ABS (x).

  \[ 178.9 \div -5.6 \equiv 14.08928571 \]

  **OPERATION:**

  \[
  \begin{array}{c}
  \text{ABS} 178.9 \div -5.6 \rightarrow 14.08928571
  \end{array}
  \]

- **INT**
  For INT (x), returns the largest integer which does not exceed the value of x.

  **OPERATION:**

  \[
  \begin{array}{c}
  \text{INT} \cdot 64.5 \rightarrow -65
  \end{array}
  \]

- **FIX**
  Returns the integer part of x for FIX (x).

  Integer part of 8000 ÷ 96.

  **OPERATION:**

  \[
  \begin{array}{c}
  \text{FIX} 8000 \div 96 \rightarrow 83
  \end{array}
  \]
- **FRAC**
  Returns the fractional part of x for FRAC (X).
  
  Fractional part of \( 8000 \div 96 \).

**OPERATION:**

\[
\text{FRAC} \quad 8000 \quad \div \quad 96 \quad \Rightarrow \quad FRAC(8000/96) = 0.33333333333
\]

- **ROUND**
  The function ROUND (X, –Y) rounds the result of X at the Yth decimal place (resulting in \( Y - 1 \) decimal places).
  
  Round result of \( 8000/96 \) to three decimal places.

**OPERATION:**

\[
\text{ROUND} \quad 8000 \quad \div \quad 96 \quad \Rightarrow \quad 83.3333333333 \quad \Rightarrow \quad \text{ROUND}(8000/96, -4) = 83.333
\]

**Decimal ↔ Sexagesimal conversions**

- **DEG:**  Sexagesimal → Decimal
- **DMSS:**  Decimal → Sexagesimal

**EXAMPLE 1:**

\[12^\circ34'56'' = 12.58222222^\circ\]

**OPERATION:**

\[
\text{DEG} \quad 12 \quad + \quad 34 \quad + \quad 56 \quad \Rightarrow \quad \text{DEG}(12.34.56) = 12.582222222\]

**EXAMPLE 2:**

\[12.3456^\circ = 12^\circ20'44.16''\]

**OPERATION:**

\[
\text{DMSS} \quad 12.3456 \quad \Rightarrow \quad \text{DMSS}(12.3456) = 12^\circ20'44.16''\]
Decimal → Hexadecimal conversions

&H: Hexadecimal → Decimal
HEXS: Decimal → Hexadecimal

EXAMPLE 1:

10_{16} = 16_{10};

OPERATION:

\[ \begin{array}{c}
&H 10 \text{ HEX} \\
&H 10 \text{ 16}
\end{array} \]

EXAMPLE 2:

1000_{10} = 3E8_{16};

OPERATION:

\[ \begin{array}{c}
&H &H 1000 \text{ HEX} \\
&H \text{ HEXS (1000 ) } \\
&H C3E8
\end{array} \]

* Hexadecimal A, B, C, D, E, F corresponds to decimal 10, 11, 12, 13, 14, 15.

FACT, NPR, NCR

These function return the factorial, permutation, and combination of entered values.

EXAMPLE 1:

10! = 3628800

OPERATION:

\[ \begin{array}{c}
&H &H 10 \text{ FACT} \\
&H \text{ FACT } 10 \\
&H 3628800
\end{array} \]

EXAMPLE 2:

10P4 = 5040

OPERATION:

\[ \begin{array}{c}
&H &H 10 \text{ 4 NPR} \\
&H \text{ NPR (10 . 4) } \\
&H 5040
\end{array} \]

EXAMPLE 3:

10C4 = 210

OPERATION:

\[ \begin{array}{c}
&H &H 10 \text{ 4 NCR} \\
&H \text{ NCR (10 . 4) } \\
&H 210
\end{array} \]
REC ↔ POL

Converts rectangular coordinates to polar coordinates, and vice versa.

EXAMPLE 1:

Convert polar coordinates \((5, \frac{\pi}{6})\) to rectangular coordinates \((X, Y)\).

OPERATION:

\[
\begin{array}{c}
\text{REC} \ (5, \frac{\pi}{6}) \quad Y \underline{\hat{E}} \\
\text{(X coordinate):} \quad 4.330127019 \\
\text{(Y coordinate):} \quad 2.5
\end{array}
\]

EXAMPLE 2:

Convert rectangular coordinates \((1, 1)\) to polar coordinates \((r, \theta)\).

OPERATION:

\[
\begin{array}{c}
P\text{OL} \ (1, 1) \quad Y \underline{\hat{E}} \\
r \underline{\hat{E}} \quad 1.414213562 \\
\theta \underline{\hat{E}} \quad 0.7853981634
\end{array}
\]
### Scientific Function Table

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Formula</th>
<th>Format</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigonometric</td>
<td>sin</td>
<td>SIN (numeric expression)</td>
<td>$-1440^\circ &lt; \text{numeric expression} &lt; 1440^\circ$ (8 rad, 1600 gradians)</td>
</tr>
<tr>
<td></td>
<td>cos</td>
<td>COS (numeric expression)</td>
<td>$-1440^\circ &lt; \text{numeric expression} &lt; 1440^\circ$ (8 rad, 1600 gradians)</td>
</tr>
<tr>
<td></td>
<td>tan</td>
<td>TAN (numeric expression)</td>
<td>$-1440^\circ &lt; \text{numeric expression} &lt; 1440^\circ$ (8 rad, 1600 gradians)</td>
</tr>
<tr>
<td>Inverse Trigonometric</td>
<td>sin$^{-1}$</td>
<td>ASN (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td>Hyperbolic</td>
<td>sinh</td>
<td>HYPSIN (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td></td>
<td>cosh</td>
<td>HYPCOS (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td></td>
<td>tanh</td>
<td>HYPTAN (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td>Inverse Hyperbolic</td>
<td>sinh$^{-1}$</td>
<td>HYPASN (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td></td>
<td>cosh$^{-1}$</td>
<td>HYPACS (numeric expression)</td>
<td>$1 \leq \text{numeric expression} &lt; 5 \times 10^{10}$</td>
</tr>
<tr>
<td></td>
<td>tanh$^{-1}$</td>
<td>HYPATN (numeric expression)</td>
<td>In numeric expression $</td>
</tr>
<tr>
<td>Exponential</td>
<td>$e^x$</td>
<td>EXP (numeric expression)</td>
<td>$-227 \leq \text{numeric expression} \leq 230.2585092$</td>
</tr>
<tr>
<td>Natural logarithm</td>
<td>$\log_{10} x$</td>
<td>LN (numeric expression)</td>
<td>numeric expression $&gt; 0$</td>
</tr>
<tr>
<td>Common logarithm</td>
<td>$\log_{e} x$</td>
<td>LOG (numeric expression)</td>
<td>numeric expression $&gt; 0$</td>
</tr>
<tr>
<td>Square root</td>
<td>$\sqrt{x}$</td>
<td>SQR (numeric expression)</td>
<td>numeric expression $\geq 0$</td>
</tr>
<tr>
<td>Cube root</td>
<td>$\sqrt[3]{x}$</td>
<td>CUR (numeric expression)</td>
<td>numeric expression $\geq 0$</td>
</tr>
<tr>
<td>Absolute value</td>
<td>$</td>
<td>x</td>
<td>$</td>
</tr>
<tr>
<td>Sign</td>
<td>$\text{SGN (numeric expression)}$</td>
<td></td>
<td>\begin{align*} { \text{numeric expression} &lt; 0 : &amp; -1 \ \text{numeric expression} = 0 : &amp; 0 \ \text{numeric expression} &gt; 0 : &amp; 1 } \end{align*}</td>
</tr>
<tr>
<td>Integer</td>
<td></td>
<td>INT (numeric expression)</td>
<td>Gauss function: Returns maximum integer value that does not exceed numeric expression value.</td>
</tr>
<tr>
<td>Fraction</td>
<td></td>
<td>FRAC (numeric expression)</td>
<td>Returns fractional part of numeric expression.</td>
</tr>
<tr>
<td>Function Name</td>
<td>Formula</td>
<td>Format</td>
<td>Details</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Rounding</td>
<td>ROUND (x, y)</td>
<td>x, y : numeric expression</td>
<td>Rounds x at position specified by y.</td>
</tr>
<tr>
<td>Fix</td>
<td>FIX (numeric expression)</td>
<td></td>
<td>Returns integer part of x.</td>
</tr>
<tr>
<td>Degree</td>
<td>DEG (d [,m[,s]])</td>
<td>d, m, s : numeric expression</td>
<td>Converts sexagesimal to decimal.</td>
</tr>
<tr>
<td>PI</td>
<td>P I</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random number</td>
<td>RAN # (numeric expression)</td>
<td></td>
<td>Returns a random number with 10 decimal places. 0 &lt; RAN # &lt; 1</td>
</tr>
<tr>
<td>Factorial</td>
<td>x !</td>
<td>FACT (numeric expression)</td>
<td>integer : 0 ≤ numeric expression ≤ 69</td>
</tr>
<tr>
<td>Permutation</td>
<td>nPr</td>
<td>NPR (n, r)</td>
<td>integer : 0 ≤ r ≤ n ≤ 10^6</td>
</tr>
<tr>
<td>Combination</td>
<td>nCr</td>
<td>NCR (n, r)</td>
<td>integer : 0 ≤ r ≤ n ≤ 10^6</td>
</tr>
<tr>
<td>Coordinate</td>
<td>POL (x, y)</td>
<td></td>
<td>Converts rectangular coordinates specified by</td>
</tr>
<tr>
<td>conversions</td>
<td>REC (r, θ)</td>
<td></td>
<td>Converts polar coordinates specified by 0 ≤ r &lt; 10^6 to rectangular coordinates.</td>
</tr>
</tbody>
</table>

* Except for ROUND, DEG, NPR, NCR, POL and REC, any values used with these functions need not be included in parentheses.

### 3-5 CALCULATIONS USING VARIABLES

Algebraic calculations can also be performed using variables. The following list of calculations, for example, becomes much easier to perform if a variable is assigned for the common term.

- \(2 \times 3.1415 + 5\) =
- \(3 \times 3.1415 + 6\) =
- \(4 \times 3.1415 + 7\) =
- \(5 \times 3.1415 + 8\) =

1. First, assign the value 3.1415 to the variable X.

\[ X \leftarrow 3.1415 \]

2. Then use the variable in place of the value for each of the calculations.

\[
\begin{align*}
2 \times X & + 5 \\
3 \times X & + 6
\end{align*}
\]

\[
\begin{align*}
11.283 \\
15.4245
\end{align*}
\]
Variables
The following rules apply to variable names for all types of variables used with the unit.

Variable names:
1. Are character strings with an upper case alphabetic character (A – Z, internal decimal code 65 – 90) or lower case alphabetic character (a – z, internal decimal code 97 – 122) in the leading (first) position. (See the character code table on page 395 for internal codes.)
2. Are composed of upper or lower case alphabetic characters or numbers (0 – 9, internal decimal code 48 – 57) following the leading alphabetic character.
3. Cannot use reserved words (see page 400) as the leading characters.
4. Can be up to 15 characters long.

3-6 OTHER CALCULATIONS
Besides the fundamental arithmetic operations of addition, subtraction, multiplication, and division, and exponential calculations the FX-850P is also capable of employing a variety of other arithmetic and relational operators.

Arithmetic Operators
The following arithmetic operators are used in formulas:

- Signs (+, −)
- Addition (+)
- Subtraction (−)
- Multiplication (×)
- Division (/)
- Power (^)
- Integer division (¥)
- Remainder of integer division (MOD)

The values used with the ¥ and MOD operators are limited to the range of −32768 through 32767, and the fractional part of non-integer values is truncated.

EXAMPLE:

5 ¥ 2.9 = 2
(5 ÷ 2.9 = \[ \frac{5}{2.9} \])
(5 ÷ 2.9 = \[ \frac{5}{2.9} \])
(The fractional parts crossed out with
"x" are truncated before the calculation
is performed.)

7 ¥ MOD 2 ¥ x = 1 * (7 ÷ 2 = 3 \cdots \frac{1}{2})

* When a variable is used instead of 7.3, a space is required between the variable and the MOD operator.

With both ¥ and MOD, the values are converted to their absolute values before division is performed. The sign assigned to the result of the ¥ operation follows the rules of normal division, while the sign assigned to the result of the MOD operation is the sign of the dividend.

EXAMPLE:

-15 ¥ 7 = -2
-15 ÷ 7 = -2 \ldots -1
-15 MOD 7 = -1
-15 ÷ 7 \equiv -15 MOD 7
Logical Operators

The application of logical operators is similar to that of arithmetic operators. The fractional parts of the data are truncated and the specified logical operation is performed bit-by-bit (each bit of the result is obtained by examining the bit in the same position for each argument). There are four different logical operators available with the unit.

**NOT**  Makes an expression not true.
**AND**  Expression is true if both parts are true, otherwise expression is false.
**OR**    Expression is true if either part is true, otherwise expression is false.
**XOR**   Expression is false if either part is true or either part is false, expression is true if one part is true and one part is false.

<table>
<thead>
<tr>
<th>X</th>
<th>NOT X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X AND Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X OR Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X XOR Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**EXAMPLE:**

Determine the logical sum for 10 and 3.

\[
\begin{array}{cc}
10 & 0 \oplus 3 \\
10 = 1010(2) & 3 = 0011(2) \\
11 = 1011(2)
\end{array}
\]

* 1010(2) represents the binary value of 10.

**Character Operator**

The only string operator available is the plus (+) operator. The length of the result is limited to 255 characters.

**EXAMPLE:**

"A" + "B" -> "AB"
Relational Operators

The following operators can be used within programs (only) to compare two values or strings. A true result returns a value of -1, while a false result returns 0.

Equal to =
Not equal to <>
Less than <
Greater than >
Less than or equal to <=
Greater than or equal to >=

With character string comparisons, each character in the string to the left of the operator is compared with each character at the corresponding position in the string to the right of the operator. Comparisons are made using the character code for each character. If two strings are of different length and the shorter string is identical to the leading characters of the longer string, the shorter string is judged to be the lesser of the two.

EXAMPLE:

10 PRINT 125 > 12
20 PRINT "DEF" < "ABCD"
30 PRINT "ABCD" = "ABC"

Since 125 is, in fact, greater than 12, a value of -1 (TRUE) is returned.

The character code of "DEF" is greater than that of "ABCD", so 0 (FALSE) is returned.

The string "ABCD" is not equal to string "ABC", so 0 (FALSE) is returned.

* Character strings are compared until a difference is found, and judgment is made upon the first difference encountered. In the above example, the "A" in the first position of one string differs from the "D" in the first position of the other string, so the comparison is based upon "A" and "D". Though the string "ABCD" is longer, string "DEF" is considered to be the greater of the two because the character code of "D" is greater than the character code of "A".
The formula storage function is very useful when performing repeat calculations. Three different keys are used when working with the formula storage function.

- key........Stores presently displayed formula.
- key........Displays formula stored in memory.
- key........Assigns values to variables in formula, and displays formula calculation result.

Sample Application

EXAMPLE:

Obtain the value of y for each of the values assigned to x when y = 3.43 cosx. (Calculate in three decimal places.)

<table>
<thead>
<tr>
<th>x</th>
<th>8°</th>
<th>15°</th>
<th>22°</th>
<th>27°</th>
<th>31°</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

OPERATION:

First specify the angle unit and number of decimal places.

- 0 4
  (Angle unit: "DEG")
- 3 8
  (Obtain in three decimal places by rounding off the 4th decimal place.)

Next, input a formula, and press the key to store it.

- 3 43 X  X

Press the key to confirm that the formula has been stored.

- 4
- 

Then, start calculating by pressing the key.
The \( \texttt{Ex} \) key can be used in place of the \( \texttt{Ex} \) key to perform repeat calculations.

* The \( \texttt{Ex} \) key can be used to terminate this function to automatically return to the CAL mode.

4-1 UTILIZATION FOR PREPARING TABLES

Multiple formulas can be written by separating with colons ( : ). Tables such as that shown below can be easily prepared by using this method.

EXAMPLE:

Complete the following table. (Calculate in three decimal places by rounding off.)

<table>
<thead>
<tr>
<th>( X )</th>
<th>( Y )</th>
<th>( P = X \times Y )</th>
<th>( Q = X / Y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.27</td>
<td>1.17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.17</td>
<td>6.48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.07</td>
<td>9.47</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.71</td>
<td>4.36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.98</td>
<td>3.62</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
OPERATION:

Specification of number of decimal places

Storing the formula

\[
\begin{align*}
X &= 4.27 \\
(\text{Calculation starts})
\end{align*}
\]

\[
\begin{align*}
Y &= 1.17 \\
(\text{X value})
\end{align*}
\]

\[
\begin{align*}
P &= 4.996 \\
(\text{Y value})
\end{align*}
\]

\[
\begin{align*}
P &= 4.996 \\
Q &= 3.650
\end{align*}
\]

Continue to input the values of X and Y in this manner, and the values of P and Q will be calculated in successive order and the table will be completed as shown below.

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>P = X \times Y</th>
<th>Q = X / Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.27</td>
<td>1.17</td>
<td>4.996</td>
<td>3.650</td>
</tr>
<tr>
<td>8.17</td>
<td>6.48</td>
<td>52.942</td>
<td>1.261</td>
</tr>
<tr>
<td>6.07</td>
<td>9.47</td>
<td>57.483</td>
<td>0.641</td>
</tr>
<tr>
<td>2.71</td>
<td>4.36</td>
<td>11.816</td>
<td>0.622</td>
</tr>
<tr>
<td>1.98</td>
<td>3.62</td>
<td>7.168</td>
<td>0.547</td>
</tr>
</tbody>
</table>

Variable names can consist of up to 15 upper case or lower case alphabetic characters. This means that variable names can be created which actually describe their contents. Remarks can also be affixed following variable names by enclosing the remarks within square brackets [ ]. Any character except for commas can be used within the remarks brackets.

EXAMPLE:

Complete the following table. (Calculate in two decimal places by rounding off.)

<table>
<thead>
<tr>
<th>Radius r (m)</th>
<th>Height h (m)</th>
<th>Volume of a cylinder ( (V_c = \pi r^2 h) ) (m³)</th>
<th>Volume of a cone ( (V_s = \frac{1}{3} V_c) ) (m³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.205</td>
<td>2.227</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.174</td>
<td>3.451</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.357</td>
<td>7.463</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
OPERATION:

RADIUS (M)?

Radius

HEIGHT (M)?

Height

CYLINDER (M3) = 10.16

Volume of a cylinder

CONE (M3) = 3.39

Volume of a cone

RADIUS (M)?

If the values of radius (r) and height (h) are input in this manner, volume (V₀) of the cylinder and volume (V₁) of the cone will be calculated successively and the table will be completed as shown below.

<table>
<thead>
<tr>
<th>Radius r (m)</th>
<th>Height h (m)</th>
<th>Volume of a cylinder (V₀ = πr²h) (m³)</th>
<th>Volume of a cone (V₁ = 1/3 V₀) (m³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.205</td>
<td>2.227</td>
<td>10.16</td>
<td>3.39</td>
</tr>
<tr>
<td>2.174</td>
<td>3.451</td>
<td>51.24</td>
<td>17.06</td>
</tr>
<tr>
<td>3.357</td>
<td>7.463</td>
<td>264.22</td>
<td>88.07</td>
</tr>
</tbody>
</table>

IMPORTANT

1. Up to 255 characters can be stored using the memory key. Storing new formula clears the currently stored formula.
2. Memory contents are retained even when power is switched OFF, either manually or by the auto power OFF function.
3. The memory key can only be used to execute numeric expressions stored using the memory key.
4. An error is generated when an entry stored by the memory key is not a numeric expression.
5. Strings and arrays are simply displayed as stored when recalled.
6. The same limitations that apply to BASIC variables apply to formula storage function variables (see page 30).
7. Calculations are terminated under the following conditions:
   - Pressing the memory key.
   - When an error is generated.
The DATA BANK function built into this unit gives it the capability to totally replace a standard notebook. For the sake of example here, the following scientific constant table will be input into the unit’s DATA BANK.

SCIENTIFIC CONSTANT TABLE

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Numeric value</th>
<th>Unit</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acceleration of free fall</td>
<td>( g )</td>
<td>9.80665</td>
<td>ms(^{-2})</td>
<td>FREE FALL</td>
</tr>
<tr>
<td>Speed of light (in space)</td>
<td>( c )</td>
<td>2.99792458 ( \times 10^8 )</td>
<td>ms(^{-1})</td>
<td>SPEED LIGHT</td>
</tr>
<tr>
<td>Planck’s constant</td>
<td>( h )</td>
<td>6.626176 ( \times 10^{-34} )</td>
<td>Js</td>
<td>PLANCK'S</td>
</tr>
<tr>
<td>Gravitational constant</td>
<td>( G )</td>
<td>6.672 ( \times 10^{-11} )</td>
<td>Nm(^2)kg(^{-2})</td>
<td>GRAVITATION</td>
</tr>
<tr>
<td>Elementary charge</td>
<td>( e )</td>
<td>1.6021892 ( \times 10^{-19} )</td>
<td>C</td>
<td>ELEMENTARY</td>
</tr>
<tr>
<td>Electron mass</td>
<td>( m_e )</td>
<td>9.109534 ( \times 10^{-31} )</td>
<td>kg</td>
<td>ELECTRON</td>
</tr>
<tr>
<td>Atomic mass</td>
<td>( u )</td>
<td>1.6605655 ( \times 10^{-27} )</td>
<td>kg</td>
<td>ATOMIC</td>
</tr>
<tr>
<td>Avogadro constant</td>
<td>( N_A )</td>
<td>6.022045 ( \times 10^{23} )</td>
<td>mol(^{-1})</td>
<td>AVOGADRO</td>
</tr>
<tr>
<td>Boltzmann’s constant</td>
<td>( k )</td>
<td>1.380652 ( \times 10^{-23} )</td>
<td>JK(^{-1})</td>
<td>BOLTZMANN'S</td>
</tr>
<tr>
<td>Molar volume of ideal gas at s.i.p.</td>
<td>( V_m )</td>
<td>2.241383 ( \times 10^{-2} )</td>
<td>m(^3)mol(^{-1})</td>
<td>IDEAL GAS</td>
</tr>
</tbody>
</table>

5-1 DATA INPUT

The MEMO mode must be entered using the operation \( \text{.scroll to} \ 5 \) to allow input of data into the DATA BANK. At this time, the display should appear as illustrated below:

Mode symbol

Record number

Cursor

37
The symbols appearing in the center of the top line of the display indicate that the current mode is the MEMO IN mode. The value on to the upper right indicates the record number, which is actually DATA BANK data line number. The record number 1 indicates that there is still no data stored. The following is the procedure to enter the constant for the acceleration of free fall:

**OPERATION:**

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>g</td>
<td>Lower case for input of g</td>
</tr>
<tr>
<td>9 (80665)</td>
<td>Numeric value input</td>
</tr>
<tr>
<td>s 2</td>
<td>Lower case for input of ms⁻²</td>
</tr>
<tr>
<td></td>
<td>Upper case for remark input</td>
</tr>
</tbody>
</table>

Multiple items (i.e. symbols and values) can be included within a line by separating them with commas. The final step of the operation is the ENTER key which writes the data into memory. This operation also causes the cursor to disappear from the display. Either press ENTER again to display the cursor at the upper left or simply enter the first character for the next record. Either procedure switches to the next record number for entry of the next item.

**OPERATION:**

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Lower case for input of C</td>
</tr>
<tr>
<td>2.99792458 8</td>
<td>Numeric value input (exponent entered using E)</td>
</tr>
<tr>
<td>s 1</td>
<td>Lower case for input of ms⁻¹</td>
</tr>
<tr>
<td></td>
<td>Upper case for remark input</td>
</tr>
</tbody>
</table>

In this example, the value used as an exponent is entered using the E key. Note that both upper case and lower case letters were used in the first two lines. Always check the display for the current mode. The indicator CAPS indicates the upper case mode, while a clear display at the CAPS position indicates lower case.

Repeat the procedures outlined above until all ten constants are stored, and then switch to the CAL mode by pressing 8 8.

**5-2 DATA DISPLAY**

All of the data stored can now be displayed to check for proper input. While in the CAL mode, press the ENTER key to display records 1 and 2. Note here that only record 1 is displayed if its length exceeds 32 characters.

```
| 8.9.80665.ms-2 FRE E FALL |
| c.2.99792458E8.m-1 SPEED LIGHT |
```

Pressing or ENTER at this time displays records 2 and 3. Pressing or at any time displays the preceding record.

```
| c.2.99792458E8.m-1 SPEED LIGHT |
| h.6.626176E-34.Js PLANCK'S |
```

* Note that the scrolling key operations noted above differ as follows:

<table>
<thead>
<tr>
<th>OPERATION</th>
<th>RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scrolls one record down.</td>
</tr>
<tr>
<td></td>
<td>Scrolls one record up.</td>
</tr>
<tr>
<td></td>
<td>Scrolls one line up.</td>
</tr>
<tr>
<td></td>
<td>Scrolls one line down.</td>
</tr>
</tbody>
</table>
5-3 DATA EDITING

Editing of stored data is performed in the MEMO IN mode (MEM). Of course, data may also be changed during the input procedure (before MEM is pressed) by moving the cursor to the desired location using the cursor keys ( and keys) and then entering the correct data.

The following procedure is used to edit data which has already been stored.

1. Press MEM (cursor not displayed)
2. Press MEM.
3. Locate record to be changed in the first line of the display.
4. Press or to display cursor (EDIT appears on display to indicate EDIT mode).
5. Move cursor to desired location and enter correct data.
6. Press MEM (EDIT disappears from display).

The following example assumes that an error is discovered in RECORD 5 (elementary charge) during display in the CAL mode.

```
Caps CAL DEG MEM

1.6.21892E-19. C ELEMENTARY
me.9.109534E-31. kg ELECTRON
```

Press MEM 9 to enter the MEMO IN mode.

```
Caps CAL DEG MEM

1.6.21892E-19. C ELEMENTARY
me.9.109534E-31. kg ELECTRON
```

Here, the cursor can be displayed by pressing or . At this time, the EDIT symbol also appears to indicate the EDIT mode.

```
Caps CAL DEG MEM  

1.6.21892E-19. C ELEMENTARY
```

Cursor

Move the cursor to the desired location and enter the correct data. Finally, press MEM to complete the procedure (EDIT disappears from display).

```
Caps CAL DEG MEM

1.6021892E-19. C ELEMENTARY
```

```
Caps CAL DEG MEM

1.6.021892E-19. C ELEMENTARY
me.9.109534E-31. kg ELECTRON
```

39
5-4 ADDING RECORDS

New records can be added to previously input records. Records can either be appended to the end of existing records, or inserted between two existing records.

5-4-1 Data Append

1. Press [9]. Unit standing by for input of next successive record following previously stored records.
2. Enter data to append new record.

5-4-2 Data Insert

1. Press [9].
3. Use [RS] or [RS] to display existing record to follow newly inserted record.
4. Enter data.
5. Press [RS] to complete procedure.

The following example describes how to enter a record containing the constant for the absolute temperature of water at 0°C between record 8 (Avogadro constant) and record 9 (Boltzmann’s constant).

<table>
<thead>
<tr>
<th>* Name</th>
<th>Symbol</th>
<th>Numeric values</th>
<th>Unit</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute temperature of water at 0°C</td>
<td>T0</td>
<td>273.15</td>
<td>K</td>
<td>ABS TEMP</td>
</tr>
</tbody>
</table>

Enter the above data after displaying RECORD 8 on the first line of the display.

OPERATION:

[15] [273] [10] [8] [5] [A] [8] [S] [T] [E] [M] [P] [RS]

The result of the operation is as follows:

RECORD 8  Avogadro constant
RECORD 9  Absolute temperature of water at 0°C
RECORD 10 Boltzmann’s constant
RECORD 11 Molar volume of ideal gas at s.t.p

* To insert new data into RECORD 1, press [RS] after displaying RECORD 1, and then enter data for record 1. At this time, all following records are shifted downwards.
5-6-1 Conditional Search

Conditional search makes it possible to designate a specific letter, value, or word (up to eight characters long) in order to quickly locate a desired record within a large file. Entering <object data> displays the first data item in which the <object data> appears immediately following a comma. Each subsequent press of \( \equiv \) displays the following data item which contains the <object data> following a comma.

In the following example, enter \( \equiv \equiv \) to locate the Avogadro constant.

| G 6.672E-11 Nm2kg-2 GRAVITATION | e 1.6021792E-19 C ELEMENTARY |

The first record to appear is record 4 (gravitational constant) because it contains the letter N following a comma. Press \( \equiv \) again to display the next data item which satisfies the stated condition.

| Gs 6.022045E23 mo1 1 AVOGADRO | k 1.38066E-23 JK-1 BOLTZMANN'S |

Here, the desired data item is located. Of course, the \( \equiv \) key can be pressed as many times as desired until the <object data> are located. If none of the records contained the specified <object data> the cursor is displayed and the unit stands by for further input.

5-7 USING DATA BANK DATA IN PROGRAMS

Data stored within the DATA BANK can also be accessed from a BASIC program using the following program commands.

READ \( \# \)

The standard READ command is generally used to read DATA statements contained within a program. The READ \( \# \) command, on the other hand, reads data from the DATA BANK. Data are read in units from the beginning of a group of data up to the next comma.

FORMAT: READ \( \# \) variable name [, variable name]

As shown above, multiple variable names can be specified, with variable names being separated by commas. As with the standard READ command, numeric data can only be assigned to numeric variables, and string data to string variables. Mismatching variable types results in a TM error, and executing the READ \( \# \) command when data do not exist produces a DA error. Any leading spaces in a group of data are skipped, unless the group is included within quotation marks.
5-5 DATA DELETE AND ALL CLEAR

5-5-1 Data Delete

The following procedure is used to delete specific records from previously stored data.

1. Press [5].
3. Press [5] and recall record number to be deleted.
4. Press cursor key ( or ) to display EDIT symbol.
5. Press and to delete currently displayed record. All following records are shifted upwards.

5-5-2 Data All Clear

Data bank contents are retained when the power of the unit is switched off and when the NEW, NEW ALL and CLEAR commands are executed. The following procedure is used to clear all current contents of the data bank.

1. Press [1] to enter the BASIC mode.
2. Enter [N E W = ] to execute NEW# command and clear all data stored in DATA BANK.

IMPORTANT
Data cleared using the procedures outlined above cannot be recovered. Only delete or clear data when it is no longer required.

5-6 DATA SEARCH

Pressing the [5] key while in the CAL mode or MEMO IN mode displays record 1. Now, each press of the [5] key shifts the cursor to the data to the right of the next comma following the current cursor position.

The [5] key can also be used to shift the cursor to the next data item, and [5] [5] can be used to shift to the previous data item.
As with the standard RESTORE command, RESTORE # can be used to designate a specific position from which the READ # operation is to be performed.

FORMAT:  RESTORE #

Simply executing RESTORE # specifies that the next READ # or WRITE # operation is to be performed at the beginning of data currently stored in the DATA BANK.

FORMAT:  RESTORE # “object string”

Including an object string with the RESTORE # command specifies that the next READ # or WRITE # operation is to be performed from the data item which begins with the specified object string contained in the DATA BANK. A DA error is generated when the specified object string does not exist. The maximum capacity for a WRITE # operation is 255 characters, and exceeding this value results in an error.

FORMAT:  RESTORE # “object string”, 0

The above format is identical to RESTORE # “object string”.

FORMAT:  RESTORE # “object string”, 1

The above format specifies that the next READ # or WRITE # operation is to be performed from the record which begins with the specified object string.

FORMAT:  RESTORE # “object string”, [0 or 1], [line number or ≠ program area number]

The above format designates a jump to the specified line number or program area number for the next READ # or WRITE # operation when the specified object string does not exist.

WRITE #

The WRITE # command is used within a program to rewrite or delete DATA BANK data.

FORMAT:  WRITE # DATA BANK data

The above format replaces existing data items with the specified DATA BANK data, starting from the current READ #/WRITE # position. In the case that data A, B, C exist in the DATA BANK, with data B specified for the next READ #/WRITE # operation, executing WRITE # Y, Z results in the DATA BANK data file being changed to A, Y, Z. Executing WRITE # “Y, Z” results in A, Y, Z, C. The data line specified for the next READ #/WRITE # operation is deleted when the WRITE # command is executed without specifying DATA BANK data.
5-8 DATA BANK FUNCTION APPLICATIONS

The data bank function can be used to perform a variety of tasks in addition to the applications outlined in this section of the manual. Virtually any data imaginable can be stored.

Example:
- Formula storage function can be used in combination with DATA BANK to store, recall and execute formulas whenever they are needed.

<table>
<thead>
<tr>
<th></th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>V = 4 * P1 * R^3 / 3</td>
</tr>
<tr>
<td>2</td>
<td>S = P1 * R^2</td>
</tr>
<tr>
<td>3</td>
<td>Y = 3 * X^2 + 4</td>
</tr>
<tr>
<td>4</td>
<td>Z = S1NX - COSY</td>
</tr>
<tr>
<td>5</td>
<td>A = Z * 1.13 - X * 1.24</td>
</tr>
</tbody>
</table>

The five formulas listed above are stored in the DATA BANK. Recall the third formula, transfer it to the formula storage function, and then execute it (in the CAL mode).

<table>
<thead>
<tr>
<th></th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Displays 1st formula.)</td>
</tr>
<tr>
<td>2</td>
<td>(Displays 2nd formula.)</td>
</tr>
<tr>
<td>3</td>
<td>(Displays 3rd formula.)</td>
</tr>
<tr>
<td>4</td>
<td>(Stores 3rd formula in memory.)</td>
</tr>
<tr>
<td>5</td>
<td>(Executes formula stored in memory.)</td>
</tr>
</tbody>
</table>

Important
Note that DATA BANK record lines are limited to 255 characters. Care should be exercised when making changes using the WRITE # command not to exceed this limit. Doing so results in an error.
Standard BASIC is employed as the programming language for this unit, and this section covers application of the BASIC language.

6-1 FEATURES OF BASIC

1. BASIC is much easier to use than other programming languages such as FORTRAN, making it suitable even for novices.
2. Writing programs is also easier because program creation, editing and execution are all performed by interacting with the computer itself.

The following functions are also available:

1. High-precision calculations are made possible by display of numeric values with 10-digit mantissas and 2-digit exponents (13-digit mantissa and 2-digit exponent for internal operations).
2. A wide selection of built-in functions makes operation easier.
   ① Standard mathematical functions
       SIN COS TAN ASN ACS ATN LOG LN EXP SQR ABS SGN
       INT FIX FRAC PI ROUND RAN ≠ DEG
   ② Powerful string handling functions
       CHR$ STR$ MIDS LEFT$ RIGHTS HEX$ DM$ ASC VAL LEN
   ③ High level mathematical functions
       POL REC NCR NPR HYP$ SIN HYP$ COS HYP$ TAN HYP$ ASN HYP$ ACS
       HYP$ TAN CUR
3. 10 independent program areas
   Up to ten programs can be stored independently in memory at the same time (P0 – 9).
4. Extended variable names
   Variable names up to 15 characters long can be used, making it possible to use names that make contents easy to understand.
5. Powerful debugging function
   A TRON command displays the number of the program line currently being executed, making it possible to easily trace execution and locate mistakes in programming.
6. Powerful screen editor
   Programs can be easily modified and corrected on the screen.
7. Virtual screen function
   Though the actual physical display of the unit has a 32-column × 2-line capacity, the virtual screen is 32 columns × 8 lines. The virtual screen can be easily scrolled using the cursor keys.
8. Expanded file management
Such standard BASIC commands as OPEN, CLOSE, INPUT # and PRINT # are all available for data file reading and writing.

6-2 BASIC PROGRAM CONFIGURATION

5-2-1 BASIC Program Format
The following is a typical BASIC program which calculates the volume of a cylinder.

**EXAMPLE:**

```
10 REM CYLINDER
20 R = 15
30 INPUT "H = " ; H
40 V = PI * R ^ 2 * H  (PI indicates π)
50 PRINT "V = " ; V
60 END
```

As can be seen, the BASIC program is actually a collection of lines (six lines in the above program). A line can be broken down into a line number and a statement.

<table>
<thead>
<tr>
<th>Line</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>R = 15</td>
</tr>
</tbody>
</table>

Computers execute programs in the order of their line numbers. In the sample program listed above, the execution sequence is 10, 20, 30, 40, 50, 60. Program lines can be input into the computer in any sequence, and the computer automatically arranges the program within its memory in order from the smallest line number to the highest. Lines can be numbered using any value from 1 through 65535.

<table>
<thead>
<tr>
<th>Input sequence</th>
<th>Memory contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 R = 15</td>
<td>10 REM CYLINDER</td>
</tr>
<tr>
<td>40 V = PI * R ^ 2 * H</td>
<td>20 R = 15</td>
</tr>
<tr>
<td>60 END</td>
<td>30 INPUT &quot;H = &quot; ; H</td>
</tr>
<tr>
<td>10 REM CYLINDER</td>
<td>40 V = PI * R ^ 2 * H</td>
</tr>
<tr>
<td>30 INPUT &quot;H = &quot; ; H</td>
<td>50 PRINT &quot;V = &quot; ; V</td>
</tr>
<tr>
<td>50 PRINT &quot;V = &quot; ; V</td>
<td>60 END</td>
</tr>
</tbody>
</table>

Input sequence  Memory contents
Following the line number is a statement or statements which actually tell the computer which operation to perform. The following returns to the sample program to explain each statement in detail.

10 REM CYLINDER.............REM stands for "remarks". Nothing in this line is executed.
20 R = 15 .....................Assigns the constant 15 (radius) to variable R.
30 INPUT "H = " ; H ..........Displays H ? to prompt a value input for height.
40 V = PI * R * 2 * H ......Calculates volume (V) of cylinder.
50 PRINT "V = " ; V ........Prints result of line 40.
60 END ......................Ends program.

As can be seen, a mere six lines of programming handles quite a bit of data. Multiple BASIC program lines can also be linked into a single line using colons.

EXAMPLE:

100 R = 15: A = 7: B = 8

Such a program line is known as a "multistatement".

Details concerning the other operations contained in the above program can be found in other sections of PART 6.

6-3 BASIC PROGRAM INPUT

6-3-1 Preparation

First switch the power of the computer ON. At this time, the display should appear as illustrated below.

```
|caps| calc| dec|
---|
```

This is the CAL mode, so the operation `Esc T` should first be performed to allow input of BASIC programs. The display should now appear as illustrated below.

```
P 0 1 2 3 4 5 6 7 8 9 3536B
|caps| basic| dec|
```

Note that the indicator CAL has been replaced by BASIC to indicate the BASIC mode. This is the mode used for BASIC program input. The other indicators on the display in the BASIC mode have the following meanings.

```
P : Program area
0-9 : Program area numbers. The numbers of program areas which already contain programs are replaced by asterisks.
```
EXAMPLE:
Program stored in area 3

3536B: Capacity (number of bytes) remaining in area for writing programs and data (free area). The value will be 3536B when there are no programs or data stored in memory, with this value decreasing as storage space is used.

Ready P0: Current program area = area 0. The current program area can be switched by pressing followed by the desired program area.

EXAMPLE:
Switching to program area 5

Previously stored programs can be deleted using one of two different procedures.

NEW: Deletes program stored in current program area only.
NEW ALL: Clears all programs stored in memory.

EXAMPLE:

NEW ALL

This operation clears all programs stored in memory and returns to current program area to 0.

6-3-2 Program Input

The following input procedure inputs the sample program for calculation of the volume of a cylinder (page 46).

Note that the key is pressed at the end of each line. A program line is not entered into memory unless the key is pressed.
ONE-KEY INPUT
The one-key BASIC commands help to make program input even easier.

EXAMPLE:
Line 30 input.

6-3-3 Program Editing
The procedure used for making corrections or changes to a program depends upon what step of program input the changes are to be made.

1. Changes in a line before **RUN** is pressed
2. Changes in a line after **RUN** is pressed
3. Changes within a program already input
4. Changes within a program following the EDIT command

1. Changes in a line before **RUN** is pressed

EXAMPLE:
20  E = 15 mistakenly input for 20  R = 15

```
10 REM CYLINDER
20 E = 15
```

(Move cursor to E)

```
10 REM CYLINDER
20 R = 15
```

(Input correct character)

```
20 R = 15
```

(Editing complete)

Note that once the desired changes are made, the **RUN** key must be pressed to store the entered line into memory.

2. Changes in a line after **RUN** is pressed

EXAMPLE:
40  V = PI * R^2 * H mistakenly input for 40  V = PI * R^2 * H

```
40 V = PI * R^2 * H
```

(Move cursor to 1)

```
40 V = PI * R^2 * H
```

(Input correct character)

```
40 V = PI * R^2 * H
```

(Editing complete)

Again, the **RUN** key must be pressed to store the corrected line into memory after changes are made.

Procedures 1 and 2 show the procedures for simple exchanges of one character for another.
Characters can also be inserted and deleted using the following procedures.

i) Insert

40 \( V = P1^*R2^*H \) mistakenly input for 40 \( V = P1^*R^2*H \)

\[
\begin{align*}
40 & \ V = P1^*R2^*H_- \\
40 & \ V = P1^*R^2*H \\
40 & \ V = P1^*R^2*H \\
- & \ 
\end{align*}
\]

(Move cursor to locatio
(open one space)
(Input correct character
and press \( \mathbf{ES} \))

ii) Delete

40 \( V = P1^*R R^2*H \) mistakenly input for 40 \( V = P1^*R^2*H \)

\[
\begin{align*}
40 & \ V = P1^*R R^2*H_- \\
40 & \ V = P1^*R^2*H \\
40 & \ V = P1^*R^2*H \\
- & \ 
\end{align*}
\]

(Move cursor to charac-
(Delete character)
(Editing complete)

The \( \mathbf{ES} \) key works rather similarly to the \( \mathbf{DEL} \) operation. The difference between the two operations is as follows.

**Difference Between \( \mathbf{INS} \), \( \mathbf{DEL} \), and \( \mathbf{ES} \)**

* \( \mathbf{INS} \), \( \mathbf{DEL} \)
  Deletes the character at the current cursor location and shifts everything to the right of the cursor one space to the left.

  \[
  \begin{align*}
  A B C D E F G & \rightarrow \mathbf{INS} \rightarrow A B C E F G \\
  \text{Cursor} & \\
  \end{align*}
  \]

* \( \mathbf{ES} \)
  Deletes the character to the left of the current cursor location and shifts everything from the cursor position right one space to the left.

  \[
  \begin{align*}
  A B C D E F G & \rightarrow \mathbf{ES} \rightarrow A B D E F G \\
  \end{align*}
  \]
Changes within a program already input

The LIST command displays the program stored in the current program area from beginning to end.

```
10 REM CYLINDER
20 R = 15

END
Ready P0
```

The last line of the program is displayed when the LIST operation is complete.

```
10 REM CYLINDER
20 R = 15
```

The 8-line virtual screen of the computer now makes it possible to use the cursor keys to scroll to preceding lines not shown on the display (see page 7).

```
  10 REM CYLINDER
  20 R = 15
  30 INPUT "H = " ; H
  40 V = PI*R^2*H
  50 PRINT "V = " ; V
  60 END
Ready P0
```

When a program greater than eight lines is stored in memory, the LIST operation should be performed by specifying the line numbers to be displayed.

AMPLE:
Playing from line 110 to line 160 on the virtual screen.

```
T 110 — 160
```

```
LIST 110—160
110 A = 1
120 FOR I = 1 TO 100
130 B = LOG ( I )
140 PRINT B
150 NEXT I
160 E = A*B
Ready P0
```

Changes can be made in a program displayed by the LIST operation by using the same processes outlined for 1 and 2 above.

The key can be used to terminate the LIST operation. The key suspends the operation, and listing can be resumed by pressing .
4. Changes within a program following the EDIT command
The EDIT command makes it easier to edit or review programs already stored in memory.

EDIT EX

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>REM CYLINDER</td>
</tr>
<tr>
<td>20</td>
<td>R=15</td>
</tr>
</tbody>
</table>

From this display, $\text{<P>}$ (or $\text{<P>}$ ) advances to the following line, while $\text{<P>}$ (or $\text{<P>}$ ) returns to the previous line.

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0</td>
<td>INPUT $H=:H$</td>
</tr>
<tr>
<td></td>
<td>$V=\pi R^2 H$</td>
</tr>
</tbody>
</table>

Here, a correction will be made in line 40.

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>$V=\pi R^2 H$</td>
</tr>
<tr>
<td>50</td>
<td>PRINT $V=\cdot V$</td>
</tr>
</tbody>
</table>

(Displays line 40 at upper line of display)

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>$V=\pi R^2 H$</td>
</tr>
</tbody>
</table>

(Enables program editing)

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>$V=\pi R^2 H$</td>
</tr>
<tr>
<td></td>
<td>$V=\cdot V$</td>
</tr>
</tbody>
</table>

(Correction)

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>$V=\pi R^2 H$</td>
</tr>
<tr>
<td>50</td>
<td>PRINT $V=\cdot V$</td>
</tr>
</tbody>
</table>

($\text{<P>}$ key exits EDIT mode)

6-4 BASIC PROGRAM EXECUTION

6-4-1 Program Execution

Once a BASIC program is stored in memory, it can be executed using one of the two following procedures.

1. Using $\text{<PRG>}$ (program area) in CAL mode

EXAMPLE: $\text{<PRG> 9}$
Executes the program in program area 9.

2. Entering RUN command in BASIC mode

EXAMPLE: RUN EX
Executes the program in the current program area.

Execute the program input in the previous section to determine the volume of a cylinder with a height of 10 (radius fixed as 15).

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUN</td>
<td>EX</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>EX</td>
</tr>
<tr>
<td></td>
<td>H=710</td>
</tr>
<tr>
<td></td>
<td>V=7068.563471</td>
</tr>
</tbody>
</table>

(Executes program)

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Corresponding volume displayed when height is entered.)
Display of the volume when this program is executed causes the STOP symbol to appear in the upper right of the display. This symbol indicates that program execution has been suspended because of execution of the PRINT command. Program execution can be resumed at this time by pressing the \texttt{EO} key again. Ending a PRINT statement with a semicolon causes execution to continue when the PRINT statement is executed, causing the display of the next PRINT statement to appear immediately following the previous display.

\textbf{Example 1:}

\begin{verbatim}
10 PRINT "BASIC"
20 PRINT "PROGRAM"
30 END
\end{verbatim}

Execution of this program results in the following display.

\begin{tabular}{|c|c|}
\hline
\texttt{RUN} & \texttt{BASIC} \\
\hline
\texttt{3} & \texttt{BASIC PROGRAM} \\
\hline
\texttt{3} & \texttt{PROGRAM Ready P0} \\
\hline
\end{tabular}

\textbf{Example 2:}

\begin{verbatim}
10 PRINT "BASIC";
20 PRINT "PROGRAM"
30 END
\end{verbatim}

Including a semicolon at the end of the first PRINT statement produces the following display.

\begin{tabular}{|c|c|}
\hline
\texttt{RUN} & \texttt{BASIC PROGRAM} \\
\hline
\texttt{BASIC PROGRAM} & \texttt{Ready P0} \\
\hline
\end{tabular}

\section*{4-2 Errors}

Sometimes, the results produced by a program are not what is expected. Such irregular executions can be broadly divided under two major classifications.

- Executions that produce errors
- Simple programming errors
- Program logic errors
- Irregular execution that do not produce errors
- Mostly program logic errors

\textbf{Executions that produce errors}

\textbf{Simple programming errors}

This is the most common type of program error and is generally caused by mistakes in program syntax. Such errors result in the following message being displayed:

\begin{verbatim}
error P0—10
\end{verbatim}

This message indicates that a syntax error has been detected in line 10 of the program stored in program area 0. The indicated program line should be checked and corrected to allow proper execution.
ii) Program logic errors

This type of error is generally caused by such illegal operations as division by zero or LOG(0). Such errors result in the following message being displayed:

```
MA error P0—10
```

As before, this message indicates that a mathematical error has been detected in line 10 of the program stored in program area 0. In this case, however, program lines related to the indicated program line as well as indicated program line itself should be examined and corrected. When an error message is displayed, the following two operations can be used to display the line number in which the error was detected.

```
| LIST 10 | EDIT 10 |
```

* The periods contained in LIST . and EDIT . instruct the computer to automatically display the last program line executed.
* Change to the BASIC mode if a BASIC program was executed in the CAL mode.

2. Irregular execution that do not produce errors

Such errors are also caused by a flaw in the program, and must be corrected by executing the LIST or EDIT command to examine the program to detect the problem. The TRON command can also be used to help trace the execution of the program. Entering TRON EXIT causes the TR symbol to appear on the display to indicate that the trace mode is ON. Now executing a BASIC program displays the program area and line number as execution is performed, and halts execution until EXIT is pressed. This allows confirmation of each program line, making it possible to quickly identify problem lines. Executing TROFF EXIT cancels the trace mode.

6-5 COMMANDS

Though there are a variety of commands available in BASIC for use in programs, the nine fundamental commands listed below are the most widely used.

The following program reads data items contained within the program itself, with the number of data items read being determined by input from an operator. The operator may input any value, but note that values greater than 5 are handled as 5 (because there are only 5 data items in line 180). The program then displays the sum of the data read from line 180, followed by the square root and cube root of the sum. Program execution is terminated when a zero is entered by the operator.

```
10 S = 0                      ........... Clears current total assigned to S
20 RESTORE                      ......... Specifies read operation should begin with first data item
30 INPUT N                      ........... Input of number of data items to be read
40 IF N > 5 THEN N = 5       ........... Input of value greater than 5 should be treated as 5
50 IF N = 0 THEN GOTO 130       ........... Jump to line 130 when input is zero
60 FOR I = 1 TO N
70 READ X                      Read X          ........... This section repeated the number of times specified by operator input in line 30
80 S = S + X                    ......... Sum of data calculation
90 NEXT I
```
100 GOSUB 140.........................Branch to subroutine starting from line 140
110 PRINT S ; Y ; Z...................Displays contents of variables S, Y, Z
120 GOTO 10.........................Jump to line 10
130 END..................................Program end
140 REM SQUARE ROOT/CUBE ROOT...Remarks
150 Y = SQRT S.........................Square root calculation
160 Z = 6 * (3).........................Cube root calculation
170 RETURN..............................Return to the statement following the statement which
called the subroutine
180 DATA 9, 7, 20, 28, 36............Data read by READ statement in line 70

REM
The REM command (line 140) is actually short for the word "remarks". The computer
disregards anything following a REM command, and so it is used for such purposes as labels
in order to make the program list itself easier to follow. Note that a single quotation mark
( ) can be used in place of the letters "REM".

INPUT
The INPUT command (line 30) is used to allow input from the computer's keyboard during
program execution. The data input are assigned to a variable immediately following the INPUT
command. In the above example, input numeric data are assigned to the variable N. Note
that a string variable must be used for string input.

EXAMPLE:
0 INPUT AS (string input)

PRINT
The PRINT command (line 110) is used to display data on the computer's display. In this
example, this command is used to display the results of the sum, square root, and cube root
calculations. When the data are displayed, the STOP symbol appears and program execution
is suspended. Execution can be resumed by pressing the Esc key.

END
The END command (line 130) brings execution of the program to an end, and can be included
anywhere within a program.

IF - THEN -
The IF/THEN command (lines 40 and 50) is used for comparisons of certain conditions, basing
the next operation upon whether the comparison turns out to be true or false. Line 40
checks whether or not value assigned to N is greater than 5, and assigns a value of 5 to N
when the original value is greater. When a value of 5 or less is originally assigned to N,
execution proceeds to the next line, with N retaining its original value. Line 50, checks whether
not the value assigned to N is zero. In the case of zero, program execution jumps to line
30, while execution proceeds to the next line (line 60) when N is any other value besides zero.
Line 50 can also be abbreviated as follows:
50 IF N = 0 THEN 130
Program areas can also be specified as jump destinations:
IF A = 1 THEN GOTO #2 (Program stored in program area 2 executed when A equals 1)
GOTO
The GOTO command (lines 50 and 120) performs a jump to a specified line number or program area. The GOTO statement in line 120 is an unconditional jump, in that execution always returns to line 10 of the program whenever line 120 is executed. The GOTO statement in line 50, on the other hand, is a conditional jump, because the condition of the IF-THEN statement must be met before the jump to line 130 is made.
* Program area jumps are specified as GOTO #2 (to jump to program area 2).

FOR/NEXT
The FOR/NEXT combination (lines 60 and 90) forms a loop. All of the statements within the loop are repeated the number of times specified by a value following the word "TO" in the FOR statement. In the example being discussed here, the loop is repeated N number of times, with the value of N being entered by the operator in line 30.

READ/DATA/RESTORE
These statements (lines 70, 180, 20) are used when the amount of data to be handled is too large to require keyboard input with every execution. In this case, data are included within the program itself. The READ command assigns data to variables, the DATA statement holds the data to be read, and the RESTORE command is used to specify from which point the read operation is to be performed.
In the sample program here, the READ command reads the number of data items specified by the input for variable N. Though the DATA statement holds only five data items, the RESTORE command in line 20 always returns the next read position to the first data item, the READ statement never runs out of data to read.

GOSUB/RETURN
The GOSUB/RETURN commands (lines 100 and 170) are used for branching to and from subroutines. Subroutines (lines 140 through 170) are actually mini programs within the main program and usually represent routines which are performed repeatedly at different locations within the main program. This means that GOSUB/RETURN makes it possible to write the repeated operation once, as a subroutine, instead of writing each time it is needed within the main program.

EXAMPLE:

120 GOSUB 1000
;
370 GOSUB 1000
;

Execution of the RETURN statement at the end of a subroutine returns execution of the program back to the statement following the GOSUB command. In this sample program, execution returns to line 110 after the RETURN command in line 170 is executed.
* GOSUB routines can also be used to branch to other program areas, as in GOSUB #3 (branches to program area 3). Note, however, that a return must be made back to the original program area using the RETURN command before an END command is executed.
* See PART 10 COMMAND REFERENCE for further details on BASIC commands.
6-6 OPERATORS

The following are the operators used for calculations which involve variables.

<table>
<thead>
<tr>
<th>Operators</th>
<th>Arithmetic operators</th>
<th>Relational operators</th>
<th>Logical operators</th>
<th>String operator</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Signs</td>
<td>Equal to</td>
<td>NOT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Addition</td>
<td>Does not equal</td>
<td>Logical product</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subtraction</td>
<td>Less than</td>
<td>Logical sum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Multiplication</td>
<td>Greater than</td>
<td>EXclusive OR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>Less than or equal to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Power</td>
<td>Greater than or equal to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Integer division</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Integer remainder of integer division</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Arithmetic Operators (+, −, *, /, ^, %, MOD)

Fractions are truncated in % and MOD calculations, when the operands on both sides of the operator are not integers.

In % and MOD calculations, the division is performed with the absolute values of both operands. In integer division (%), the quotient is truncated to an integer. With the MOD operator, the remainder is given the sign of the dividend.

**EXAMPLES:**

\[-15 \div 7 = -2\]
\[-15 \mod 7 = 1\]
\[-15 \% 7 = -15 \mod 7\]

The length of an expression is limited to 255 characters.

Relational Operators ( =, <>, >, <, >, =<, <=, =>, >= )

Relational operations can be performed only when the operators are both strings or numeric values.

With strings, character codes are compared one-by-one from the beginning of the strings. This is to say that the first position of string A is compared with the first position of string B, the second position of string A with the second position of string B, etc. The result of the comparison is based upon the character codes of the first difference between the strings detected, regardless of the length of the strings being compared.
EXAMPLES:

<table>
<thead>
<tr>
<th>STRING A</th>
<th>STRING B</th>
<th>RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
<td>ABC</td>
<td>A = B</td>
</tr>
<tr>
<td>ABC</td>
<td>ABCDE</td>
<td>A &lt; B</td>
</tr>
<tr>
<td>ABC</td>
<td>XYZ</td>
<td>A &lt; B</td>
</tr>
<tr>
<td>XYZ</td>
<td>ABCDE</td>
<td>A &gt; B</td>
</tr>
</tbody>
</table>

(character code for A less than that for X)
(character code for X greater than that for A)

A result of -1 is returned when the result of a relational operation is true (conditions met), while 0 is returned when the result is false (conditions not met).

EXAMPLE:

10 PRINT 10 > 3.......................... -1 returned because 10 > 3 is true
20 PRINT 7 < 1............................ 0 returned because 7 < 1 is false
30 PRINT "ABC" = "XYZ"................. 0 returned because ABC = XYZ is false
40 END

3. Logical Operators
The operands of logical operations are truncated to integers and the operation is performed bit-by-bit to obtain the result.

<table>
<thead>
<tr>
<th>X</th>
<th>NOT X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X AND Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X OR Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X XOR Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

4. String Operators (+)
Strings may be concatenated using a + sign.
The result of the operation (including intermediate results) may not exceed 255 characters.

EXAMPLE:
AS = "AD" + "1990"
The above example results in the string "AD1990" being assigned to variable AS.
Order of Operations

Arithmetic, relational and logical operations are performed in the following order of precedence:

\[
\begin{align*}
(, ) \\
\text{Scientific function} \\
\text{Power} \\
\text{Sign (+, –)} \\
\times, /, \%, \text{MOD} \\
\text{Addition and subtraction} \\
\text{Relational operators} \\
\text{NOT} \\
\text{AND} \\
\text{OR, XOR}
\end{align*}
\]

Operations are performed from left to right when the order of precedence is identical.

7 CONSTANTS AND VARIABLES

7-1 Constants

The following shows the constants included in the sample program on page 46:

```
OGRAM        CONSTANTS
R = 15        15
INPUT "'H=' ; H " "H='" 
V = PI * R^2 * H 2 
PRINT "'V=' ; V " "V='"
```

these, 15 and 2 are numeric constants, while "'H='" and "'V='" are string constants.

Numeric Constants

Numeric Notation
Decimal notation
Hexadecimal notation

Numeric Value Precision

Internal numeric operations
12-digit mantissa, 2-digit exponent (PI = 11 digits: 3.1415926536; displayed in 10 digits: 3.141592654)
Results
10-digit mantissa, 2-digit exponent (exponent rounded to 10 digits)
Number of characters per line
255 characters per line
Result Display
Integers less than \(1 \times 10^{10}\) : Integer display
Decimal portion less than 11 digits : Decimal display
Other : Exponential display
Display rounding : Results are rounded off at the 10th digit and displayed.
String Constants

Strings within quotation marks (i.e. "ABC", "H =")
Closing quotation marks at the end of a line may be omitted (10 PRINT "TEST" can be written 10 PRINT "TEST")
Multiple strings can be connected with a " + " sign.

6-7-2 Variables

Numeric Variables

The following shows the numeric variables included in the sample program on page 46:

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>NUMERIC VARIABLES</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>R = 15</td>
</tr>
<tr>
<td>30</td>
<td>INPUT &quot;H =&quot; ; H</td>
</tr>
<tr>
<td>40</td>
<td>V = PI * R^2 * H</td>
</tr>
</tbody>
</table>

Numeric variables are so named because their contents are handled as numbers. Numeric variable names can be up to 15 characters long, and are used within programs to store calculation results or constants in memory. In the sample program, the value 15 is stored in H, while V, which is the result of the calculation, holds the value which represents the volume of the cylinder. As can be seen, assignment to a variable is performed using the " = " symbol. This differs from an equal sign in that it declares that what is to the right should be assigned to what is to the left. Actually, a variable can be thought of as a kind of box as illustrated below:

```
\[ \text{R} \]
\[ \text{V} \]
```

String Variables

Another type of variable is known as a string variable, which is used to store character string data. String variable names are indicated by "$" following the name.

EXAMPLE:

```
10 A$ = "AD"............................Assigns "AD" to string variable A$.
20 INPUT "YEAR =" ; BS............Assigns keyboard input to variable BS.
30 C$ = A$ + BS........................Assigns combination of A$ and BS to C$.
40 PRINT C$..............................Displays contents of C$.
50 END
```

In the above example program, entering a year such as 1990 in line 20 results in a display of AD1990 in line 40.

* With string variables, " + " can be used to connect two strings.

* Note here that strings cannot be assigned to numeric variables such as A, and numeric values cannot be assigned to string variables such as A$.
array Variables

numeric variables and string variables can store only one data item per variable. Beside of this, large amounts of data are better handled using array variables (usually referred simply "arrays"). Before an array variable can be used within a program, a DIM statement must appear at the beginning of the program to "declare" to the computer that an array variable is to be employed.

AMPLE:

Declare array variable A for storage of 21 data items.

DIM A (20)

The above format is used to declare "ARRAY VARIABLE NAME (NUMBER OF ELEMENTS)".

Declared value of 20 makes it possible to store 21 data items (see page 63 for details).

AMPLE:

The sum (X) and the sum of the squares (Y) for the following 10 data items:

12, 9, 11, 13, 14, 11, 12, 9, 10

The following program would be required to perform the calculation if only simple numeric variables are used:

```
X = A1 + A2 + A3 + A4 + A5 + A6 + A7 + A8 + A9 + A10......Calculates sum
Y = A1^2 + A2^2 + A3^2 + A4^2 + A5^2 + A6^2 + A7^2 + A8^2 + A9^2 + A10^2......Calculates sum of squares
```

Program becomes much simpler when an array is used.

```
DIM A (10).....Declares array
A (1) = 10 : A (2) = 12 : A (3) = 9 : A (4) = 11 : A (5) = 13
A (6) = 14 : A (7) = 11 : A (8) = 12 : A (9) = 9 : A (10) = 10
Assigns values to array
X = 0 : Y = 0
FOR I = 1 TO 10
X = X + A (I) : Y = Y + A (I)^2
NEXT I
Calculates sum and sum of squares
```

At first glance, the array may appear to be rather troublesome to use, but it actually makes programming simpler when large volumes of data are being assigned.
EXAMPLE:
100 data items

Numeric variables

Assigns values to variables

Calculates sum

Calculates sum of squares

Array

Declares array

Assigns values to array

Calculates sum and sum of squares

Data

A look at these programs reveals that an increase in data entails virtually no change in the portion which calculates the sum and sum of squares. The only changes would be in lines 10, 20, and 40, where the constant would be changed from 10 to 100.

Again, the concept of the array can be better grasped by thinking of them as boxes. Previously, a simple variable was described as a single box. Arrays, on the other hand, would be a series of numbered boxes which form a set.
Illustrated above, the array A(10) actually contains a total of eleven boxes, numbered from 0 through A(10), with each box being capable of holding a different value. The actual term used to refer to a box is "element". Recalling a stored value is performed by simply specifying the corresponding element number.

**Example:**

Value stored in element 4 of array A

\[ 4 : Y = A_4(X) \]

Value which specifies an element in an array (4 above) is called a subscript.

Now, the only arrays covered have been those formed by a single line of elements or "rows". These are known as "one-dimensional" arrays. Arrays may also contain more than one dimension with elements connected vertically and horizontally into two-dimensional and e-dimensional arrays.

**Example:**

A(2, 3)

Declaration in this example sets up an array of three lines and four columns, making it capable of storing 12 different values.

**Numeric arrays and string arrays**

With simple variables, arrays can also be declared to hold strings by using the "$" symbol following the array variable name. Again remember, numeric values cannot be assigned to string arrays and strings cannot be assigned to numeric arrays.
EXAMPLE:
The following procedure is used to declare an array and store the data for five individuals and their points scored during a certain game.

String array NS(5) declared for names
Numeric array P(5) declared for points

10 DIM NS (5), P (5)..............Declaration of arrays to store names and points
20 FOR I=1 TO 5
30 READ A$, X
40 NS ( I )= A$.......................Stores names to string array
50 P ( I )= X........................Stores points to numeric array
60 NEXT I
70 END
80 DATA SMITH, 70, BROWN, 68, JONES, 87, CARTER, 80, MILLS, 74

6-7-3 Summary

Variable Types

The three following types of variables are available for use with this unit.

1. Numeric variables (up to 12-digit mantissa) A, a, NUMBER, POINTS
2. String variables (up to 255 characters) AS, STRING$
3. Array variables ---- Numeric array A (10), XX (3, 3, 3)
   --- String array AS (10), ARRAYS$ (2, 2)

Variable Names

- Variable names can consist of upper, lower case or numeric characters, but a numeric character cannot be used in the first position of the variable name (i.e. 1AE, 3BC$ are illegal).
- Reserved words (see page 400) cannot be used as the leading characters of a variable name (i.e. RUNON, LIST1$ are illegal).
- The maximum length of a variable name is 15 characters.

Arrays

1. Arrays are declared by DIM statements.
2. Elements described by subscripts which are integers greater than 0. Fractions are disregarded.
3. The number of dimensions is limited by stack capacity.
4. The maximum value of subscripts is limited by memory capacity.

Variable/Array Application

1. Variables and arrays can be used jointly by all program areas.
2. Arrays cannot be used unless first declared using the DIM statement.
Counting Bytes Used by Variables

The following outlines the number of bytes reserved when a variable appears the first time within a program.

**Numeric Variables**

- Variable name length + 12) bytes in variable area

**String Variables**

- Variable name length + 4) bytes in variable area and (string length + 1) bytes in string area

String lengths for array variables when the array is declared by the DIM statement.

**Numeric Array Variables**

- Variable name length + 4) + (array size × 8) + (dimension × 2 + 1) bytes in variable area

**Example:**

```plaintext
1 XYZ (3, 3, 5, 2)
```

```plaintext
 ne    : 3
e     : 4 × 4 × 6 × 3 = 288
dimension : 4
calculation: (3 + 4) + (288 × 8) + 4 × 2 + 1 = 2320 bytes
```

**String Array Variables**

- Variable name length + 4) + (array size) + (dimension × 2) bytes in variable area.

Lengths of individual strings are required in the variable area when strings are assigned the array.

**Example:**

```plaintext
DIM ABS (3, 3)
```

```plaintext
ABS (0, 0) = "* * * * * *"
```

```plaintext
 ne    : 2
e     : 4 × 4 = 16
dimension : 2
calculation: (2 + 4) + 16 + (2 × 2) + 5 bytes
```

**Calculating Program Length**

The following shows points which must be considered when calculating memory requirements for programs.

- **Numbers**: 2 bytes per line number, regardless of number length (1 - 65535)
- **Commands**: 2 bytes per command
- **Functions**: 2 bytes per function
- **Numeric/Alphabetic Characters**: 1 byte per character (spaces also counted as characters)
- **Key**: 1 byte per key operation at end of program line (for storage of line)

A byte added to sum of the above.

**Example:**

```plaintext
A = SIN X
```

```plaintext
ne number) + 1 (space following line number) + 1 (A) + 1 (=) + 2 (SIN) + 1 (space)
1 (X) + 1 ( = ) + 1 = 11
```

A calculation indicates that a total of 11 bytes are required for storage of the above program. The space following the line number is added automatically.
6-8 PROGRAM SAVE AND LOAD

The following save and load procedures can only be performed when the FA-6 interface unit is used.

6-8-1 Program Save

Programs stored in the memory of the unit are protected by the memory back up battery even when the power of the unit is switched OFF. The entire contents of the memory, however, are deleted whenever both the main power supply batteries and memory back up batteries are removed from the unit at the same time, or when the NEW ALL command is executed. Program area contents can be stored onto standard cassette tapes to protect against loss of important data, or to make room for further programming when all program areas are full. The following two commands are available for such save operations.

SAVE : Saves contents of current program area.
SAVE ALL : Saves entire contents of all program areas.

EXAMPLE:
Executing SAVE in this case saves the contents of program area P0, while SAVE ALL would save the contents of program areas P0 through P9.

SAVE "P0"
(Saves program in program area 0)
SAVE "P0"
(Save complete)

Filenames up to eight characters long can also be assigned to programs stored on cassette tapes using the SAVE and SAVE ALL commands.

SAVE "BASIC"
(Saves program under filename "BASIC")
SAVE "BASIC"
(Save complete)

6-8-2 Program Verify

The VERIFY command makes it possible to verify whether or not the program saved using SAVE or SAVE ALL was copied correctly to the cassette tape.

EXAMPLE:
Verify correct save of the program BASIC

VERIFY "BASIC"
(Verification of saved program)
BASIC B
(Finds specified program and verifies)
BASIC "P0"
(Verification complete)

If the Ready prompt does not appear after some time, check whether or not the filename entered with the VERIFY command is correct. If it is correct, adjust the volume level of the cassette recorder being used and repeat the verification procedure.

PO error
Ready P0
The error message illustrated above indicates that the program was not saved correctly. In this case, check the following items:

Verify the program again, this time appending "CAS1:" before the filename (VERIFY "CAS1 BASIC" in the above example).
Ensure that connections between the computer and cassette tape recorder are correct and secure.
Ensure that the volume level of the recorder is set to in the vicinity of its maximum.
Check whether the cassette tape is damaged.
Check whether the recorder heads are soiled.

Note also that an error will be generated if a program exists on the tape with the same name as that currently present in computer memory, but the contents of the two programs are different.

The VERIFY command automatically determines whether the program being checked was saved using the SAVE or SAVE ALL command.

8-3 Program Load

Programs stored on cassette tapes using the SAVE and SAVE ALL commands can be loaded into the computer using the LOAD and LOAD ALL commands.

AMPLE:

Read the program "BASIC" from cassette tape into memory

LOAD BASIC (Program load command)

BASIC B (Program filename)

Ready P0 (Load complete)

Note that executing the LOAD and LOAD ALL commands while programs are already stored memory deletes the current memory contents.

The LOAD ALL command can be used to load programs to all of the program areas (P0 – P9). Specifying a filename in the LOAD and LOAD ALL commands causes the unit to search for the specified filename for loading into memory. The following table shows the relationship between the LOAD, LOAD ALL, SAVE and SAVE ALL commands.

<table>
<thead>
<tr>
<th></th>
<th>LOAD</th>
<th>LOAD &quot;filename&quot;</th>
<th>LOAD ALL</th>
<th>LOAD ALL &quot;filename&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAVE</td>
<td>O</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>SAVE &quot;filename&quot;</td>
<td>O</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>SAVE ALL</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>SAVE ALL &quot;filename&quot;</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

Note:
For PART 7 PERIPHERAL DEVICES for details on using the SAVE and LOAD commands.
A variety of peripheral devices are available for connection to this unit to provide even more computing power.

System Configuration

7-1 CASSETTE INTERFACE UNIT FA-6

7-1-1 Features

The FA-6 is an interface unit which makes it possible to use a cassette tape recorder as an external data storage device. Besides a cassette interface, the FA-6 is also equipped with an RS-232C interface and a Centronics standard printer interface. The RS-232C interface connector, printer interface, cassette interface and an AC adaptar jack are located on the back of the FA-6. The battery compartment is located on the bottom of the unit. Batteries are loaded by removing the battery compartment cover and inserting batteries while ensuring that their polarities (+ -) are as illustrated in the compartment.
The error message illustrated above indicates that the program was not saved correctly. In this case, check the following items:

1. Verify the program again, this time appending "CAS1:" before the filename (VERIFY "CAS1 BASIC" in the above example).
2. Ensure that connections between the computer and cassette tape recorder are correct and secure.
3. Ensure that the volume level of the recorder is set to in the vicinity of its maximum.
4. Check whether the cassette tape is damaged.
5. Check whether the recorder heads are soiled.

Note also that an error will be generated if a program exists on the tape with the same name as that currently present in computer memory, but the contents of the two programs are different.

The VERIFY command automatically determines whether the program being checked was saved using the SAVE or SAVE ALL command.

8-3 Program Load

Programs stored on cassette tapes using the SAVE and SAVE ALL commands can be loaded into the computer using the LOAD and LOAD ALL commands.

AMPLE:

Load the program "BASIC" from cassette tape into memory

```
LOAD BASIC
Ready P0
```

Note that executing the LOAD and LOAD ALL commands while programs are already stored in memory deletes the current memory contents.

The LOAD ALL command can be used to load programs to all of the program areas (P0 – P9). Specifying a filename in the LOAD and LOAD ALL commands causes the unit to search for the specified filename for loading into memory. The following table shows the relationship between the LOAD, LOAD ALL, SAVE and SAVE ALL commands.

<table>
<thead>
<tr>
<th></th>
<th>LOAD</th>
<th>LOAD &quot;filename&quot;</th>
<th>LOAD ALL</th>
<th>LOAD ALL &quot;filename&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAVE</td>
<td>O</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>SAVE &quot;filename&quot;</td>
<td>O</td>
<td>O</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>SAVE ALL</td>
<td>x</td>
<td>x</td>
<td>O</td>
<td>x</td>
</tr>
<tr>
<td>SAVE ALL &quot;filename&quot;</td>
<td>x</td>
<td>x</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>

NOTE:

See PART 7 PERIPHERAL DEVICES for details on using the SAVE and LOAD commands.
A variety of peripheral devices are available for connection to this unit to provide even more computing power.

System Configuration

7-1  CASSETTE INTERFACE UNIT FA-6

7-1-1  Features

The FA-6 is an interface unit which makes it possible to use a cassette tape recorder as an external data storage device. Besides a cassette interface, the FA-6 is also equipped with an RS-232C interface and a Centronics standard printer interface. The RS-232C interface connector, printer interface, cassette interface and an AC adaptor jack are located on the back of the FA-6. The battery compartment is located on the bottom of the unit. Batteries are loaded by removing the battery compartment cover and inserting batteries while ensuring that their polarities (±) are as illustrated in the compartment.
-2 Connections

Ensure that the power of both the computer and the interface unit is switched OFF before attempting connections. Once connected, power should be switched ON for the computer and then the interface unit.

-3 Cassette Interface

The cassette interface is used for connection of a cassette recorder to make it possible to record programs on cassette tapes and to later reload the programs into computer memory. Connection to the cassette tape recorder is accomplished using the optional SB-7 connectable cable. The red plug is inserted into the MIC or LINE IN jack (labeling differs according to type of recorder used) of the recorder, while the white plug is inserted into the EAR or E OUT jack of the recorder. The black plug is inserted into the REM jack of recorders equipped with a remote function.
NOTE:
The remote plug is not used when the recorder being used is not equipped with a remote function.
The recorder should be set to its RECORD mode when performing recording of programs or data. For program loading, set the recorder to its PLAYBACK mode after executing the LOAD command.

Single Program Save
SAVE "file descriptor" XXX (file descriptor may be omitted)
The file descriptor can contain any symbols, characters, or numbers (except quotation marks).

EXAMPLE:
SAVE "CAS0: AD1990" XXX
* "CAS0:" may be omitted.

Single Program Load
LOAD "file descriptor" XXX (file descriptor may be omitted)
If a file descriptor is not specified, the unit loads the first program found on the tape.

EXAMPLE:
LOAD "CAS0: AD1990" XXX
* "CAS0:" may be omitted.

Loading/Saving All Programs
SAVE ALL "file descriptor" XXX } (file descriptor may be omitted)
LOAD ALL "file descriptor" XXX }
The SAVE ALL command saves all of the programs stored in program areas P0 through P9 to cassette tape. LOAD ALL, on the other hand, loads programs saved using the SAVE ALL command. The LOAD ALL command also clears any contents present in the program areas and replaces them with the programs from the cassette tape.

Saving and Loading Data Bank Data
The SAVE ≠ and LOAD ≠ commands are used for the saving and loading of memo data stored in the DATA BANK. The procedure for using these commands is identical to that described for SAVE and LOAD above.

Saved File Verification
The VERIFY command checks whether the program or data saved to the cassette tape matches exactly the current memory contents.
VERIFY "file descriptor" (file descriptor may be omitted)

Tape Recorder Operation
The recorder should be set to its RECORD mode before the SAVE command is executed.
For program loading, set the recorder to its PLAYBACK mode and then execute the LOAD command.
The SAVE, SAVE ALL, LOAD, LOAD ALL, SAVE ≠, and LOAD ≠ commands cannot be used in the CAL mode or within programs. They can, however, be used during manual operations in the BASIC mode.
* See PART 9 FILE HANDLING FUNDAMENTALS for details on file descriptors.
1-4 RS-232C Interface

RS-232C Switch

The RS-232C interface can be used for data communications after the RS-232C/Cassette Switch is set to RS-232C.

Specifications

- Communication method: Start-stop (asynchronous) full-duplex mode only
- Transmission speed: 150, 300, 600, 1200, 2400, 4800 baud
- Parity bit: Odd, Even, None
- Character bit length: 7 or 8 bits
- Stop bits: 1 or 2 bits
- TS signal control: Control/no control
- SR signal control: Control/no control
- D signal control: Control/no control
- RTS control: XON/XOFF control/no control
- Transmit/output code system: SI/ST control/no control

1 Configuration

<table>
<thead>
<tr>
<th>Terminal number</th>
<th>Signal name</th>
<th>Pin connection</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FG</td>
<td></td>
<td>OUTPUT</td>
</tr>
<tr>
<td>2</td>
<td>TXD</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>3</td>
<td>RXD</td>
<td></td>
<td>OUTPUT</td>
</tr>
<tr>
<td>4</td>
<td>RTS</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>5</td>
<td>CTS</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>6</td>
<td>DSR</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>7</td>
<td>GND</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>8</td>
<td>DCD</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>9</td>
<td>NC</td>
<td></td>
<td>OUTPUT</td>
</tr>
<tr>
<td>10</td>
<td>NC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>12</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>13</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>14</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>15</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>16</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>17</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>18</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>19</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>20</td>
<td>DTR</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>21</td>
<td>NC</td>
<td></td>
<td>OUTPUT</td>
</tr>
<tr>
<td>22</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>23</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>24</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
<tr>
<td>25</td>
<td>NC</td>
<td></td>
<td>INPUT</td>
</tr>
</tbody>
</table>

See PART 9 FILE HANDLING FUNDAMENTALS for details on using the RS-232C interface.
7-1-5 Centronics Interface (Printer Interface)

General
The Centronics interface is used to output data processing results or program lists to a printer. Any Centronics printer can be connected to the computer via the FA-6 interface unit.

Pin Configuration

<table>
<thead>
<tr>
<th>Terminal number</th>
<th>Signal name</th>
<th>Pin Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PSTB</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>PDB0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>PDB1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>PDB2</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>PDB3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>PDB4</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>PDB5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>PDB6</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>PDB7</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>NC</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>BUSY</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>NC</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>NC</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>GND</td>
<td></td>
</tr>
</tbody>
</table>

BASIC Printer Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLIST</td>
<td>Outputs program contents to printer</td>
</tr>
<tr>
<td>LPRINT</td>
<td>Outputs specified characters to printer</td>
</tr>
<tr>
<td>TAB</td>
<td>Outputs spaces up to a specified position to printer</td>
</tr>
</tbody>
</table>

PRT ON Mode
Setting the unit to the PRT ON mode (7) and then executing the PRINT, LIST or VARLIST commands prints out the results of such command execution and object data specified by the TRON command on the printer. Including 7 within a program prints out all contents of subsequent print commands. The PRT ON mode can be canceled by 8 or 8 (PRT OFF).

* See the FA-6 manual for details on its proper operation.
2 PLOTTER-PRINTER (FP-100)

FP-100 is a four-color plotter-printer capable of printing on A-4 size paper.
FP-100 has both a character mode and a graphics mode which makes it possible to print virtual any type of computer output.
Character mode: Program lists, calculation results
Graphics mode: Graphics produced by graphics commands

-1 Specifications
Color printing in black, red, blue, green
Character effects: Italics
Dot resolution: 0.1mm/step
Character width: Postcard size to letter size
Character size: 1.0mm × 1.2mm (S0, 0) – 16.0mm × 19.2mm (S15, 15); 256 types

-2 Connections
A unit is connected to FP-100 via the FA-6 interface unit and PK-7 printer cable. Use only CASIO PK-7 printer cable

-3 Data Printing
Program lists are output to the printer using the BASIC LLIST command. Execution of the ST command prints out the currently accessed program.
The LPRINT command is used to print out data within a program, while execution of the ST # command in the BASIC mode ( 1 ) prints data bank contents.
See the FP-100 manual for details on its proper operation.
7-3 CHARACTER PRINTER (FP-40)

The FP-40 can be used to print out data or program lists of programs written on the FX-850P.
* Addition of an optional interface pack also makes it possible to use the FP-40 with PB-100 series and PB-700 series computers.

7-3-1 Specifications

- Print method: Thermal print system (non-impact)
- Columns: 40 standard (normal mode)
  - 80 maximum (80CHR mode)
- Print speed: Approximately 0.65 lines/sec (normal mode)
- Paper feed: 1/6 inch or 1/9 inch
- Roll paper: Width 112mm, o.d. 30mm maximum; thermal paper (TRP-112)

7-3-2 Connection

This unit is connected to the FP-40 via the FA-6 interface unit, PK-7 printer cable, and SB-43 interface pack.

* See the FP-40 manual for details on its proper operation.
RAM EXPANSION PACK (RP-8 (8KB)), (RP-33 (32KB))

The unit comes equipped with a standard RAM of 8K bytes. RAM expansion packs are also
available for larger programs and for handling larger quantities of data. Addition-
ally, the RP-8 RAM pack expands memory capacity to 16K bytes, while the RP-33 RAM pack
provides memory to 40K bytes.

-1 Expanded Memory Map

![Diagram of Expanded Memory Map]

- SYSTEM AREA
- STRING VARIABLE DATA
  - VARIABLE FREE AREA
    - (Capacity checked by FRE0)
  - NUMERIC VARIABLE DATA
  - VARIABLE TABLE
    - P0 AREA
    - P1 AREA
    - P9 AREA
  - DATA BANK AREA
    - FREE AREA
      - (Capacity checked by FRE1)
  - CONTROL AREA

1536 bytes with RP-8
8192 bytes with RP-33

11728 bytes with RP-8
29648 bytes with RP-33
7-4-2 Handling RAM Packs

Preparation
Static electrical charges can damage internal circuitry of RAM packs. Be sure to touch a
door knob or some other metal fixture to discharge static electricity before handling RAM
packs.

Procedure
① Switch the power of the unit OFF.
② Remove the back cover of the unit after removing the two screws holding it in place.
③ Insert the RAM pack into the socket provided in the unit and fasten it in place using the three
screws provided.
* Never touch the RAM pack connector or PCB pad.
④ Replace the back cover of the unit and fasten it in place using the two screws.
⑤ Switch the power of the unit ON and press the P button. Next, press the ALL RESET button and
switch the power of the unit OFF and then ON again.

* Failure to press the P button and ALL RESET button after inserting or removing the RAM
pack can result in altered unit memory contents.
* Dirt, dust, or finger prints on the RAM pack connector or PCB pad can result in poor con-
nection and malfunction. Never touch connectors.
* Be sure to store RAM packs in their original cases when removed from the computer and
store in an area free of dirt or dust.
PART 8
PB-100 SERIES COMPATIBILITY

The PB-100 series is capable of loading data and programs written for the PB-100 series computers of executing PB-100 programs. Certain special commands are required, however, to allow program compatibility.

The PB-series includes the following models:

1 PB-100 SERIES PROGRAM INPUT/EDITING

The following command conversions are required to allow execution of PB-100 series programs on this unit:

<table>
<thead>
<tr>
<th>PB-100 SERIES</th>
<th>FX-850P</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT, STATLIST, STATCLEAR</td>
<td>Use library function or create a program.</td>
</tr>
<tr>
<td>XX, EOY</td>
<td></td>
</tr>
<tr>
<td>JT, GET</td>
<td>Change using OPEN &quot;CAS0:&quot;</td>
</tr>
<tr>
<td>Example 1: PUTS—OPEN &quot;CAS0:&quot; FOR OUTPUT AS #1: PRINT #1, AS:CLOSE</td>
<td></td>
</tr>
<tr>
<td>Example 2: GETS—OPEN &quot;CAS0:&quot; FOR INPUT AS #1: INPUT #1, AS:CLOSE</td>
<td></td>
</tr>
<tr>
<td>&lt; = . = &lt;</td>
<td></td>
</tr>
<tr>
<td>&gt; &lt; , &lt; &gt;</td>
<td></td>
</tr>
<tr>
<td>&gt; = . = &gt;</td>
<td></td>
</tr>
<tr>
<td>^</td>
<td>PI</td>
</tr>
</tbody>
</table>

It is recommended that the following command conversions also be performed to ensure compatibility between PB-100 series programs and FX-850P programs.

<table>
<thead>
<tr>
<th>PB-100 SERIES COMMANDS</th>
<th>FX-850P COMMANDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>/AC</td>
<td>CLEAR</td>
</tr>
<tr>
<td>F - -</td>
<td>IF - THEN -</td>
</tr>
<tr>
<td>CSR</td>
<td>LOCATE or TAB</td>
</tr>
<tr>
<td>KEY, KEYS</td>
<td>INKEYS</td>
</tr>
<tr>
<td>RND</td>
<td>ROUND</td>
</tr>
<tr>
<td>MID (location, number of characters)</td>
<td>MIDS ($, location, number of characters)</td>
</tr>
<tr>
<td>GOTO (numeric expression), GOSUB (numeric expression)</td>
<td>ON - GOTO, ON - GOSUB</td>
</tr>
<tr>
<td>MODE 4/5/6</td>
<td>ANGLE 0/1/2</td>
</tr>
<tr>
<td>MODE 7/8</td>
<td>LPRINT</td>
</tr>
</tbody>
</table>
Though direct input of PB-100 series characters is not possible with this unit, they can be displayed using the CHRS function.

EXAMPLE:

Display ≤
PRINT CHRS (&HE1)

See CHARACTER CODE TABLE on page 395 for details on character codes.

NOTES
- A space must be included before the THEN of an IF – THEN statement when the character preceding the THEN is alphabetic.
  Example: IF3 = THEN \rightarrow IF3 = A THEN
- The jump destination of a GOTO or GOSUB statement must be enclosed in parentheses when it is an expression beginning with a numeric value.
  Example: GOTO 10*A \rightarrow GOTO (10*A)
- A space must be included before the TO in the TO portion of a FOR – NEXT loop when the initial value is represented by an alphabetic character.
  Example: FORA = BTOC \rightarrow FORA = B TOC
- An INPUT statement followed by a comma will not produce a question mark as a prompt. The question mark is displayed only when the INPUT statement is followed by a semicolon.

8-2 PB-100 SERIES PROGRAM EXECUTION

8-2-1 DEFM Mode

The PB-100 series uses variables A – Z and AS – ZS as arrays. The DEFM mode can be used when executing PB-100 series programs on this unit to use A – Z and AS – ZS in the same manner as PB-100 series computer. D F E M number of variables to be expanded as performs variable expansion as PB-100 series computers.

PB-100 series programs which use DEFM arrays should always be executed in the DEFM mode.
The variables for this unit become as follows when DEFM mode arrays are used:

<table>
<thead>
<tr>
<th>A (0)</th>
<th>B (0)</th>
<th>→ A</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (1)</td>
<td>B (1)</td>
<td>→ B</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>A (25)</td>
<td>B (24)</td>
<td>→ Z</td>
</tr>
<tr>
<td>A (26)</td>
<td>B (25)</td>
<td>→ A (0)</td>
</tr>
<tr>
<td>A (27)</td>
<td>B (26)</td>
<td>→ A (1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>A$ (0)</td>
<td>B$ (0)</td>
<td>→ A$</td>
</tr>
<tr>
<td>A$ (1)</td>
<td>B$ (1)</td>
<td>→ B$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>A$ (25)</td>
<td>B$ (24)</td>
<td>→ Z$</td>
</tr>
<tr>
<td>A$ (26)</td>
<td>B$ (25)</td>
<td>→ A$ (0)</td>
</tr>
<tr>
<td>A$ (27)</td>
<td>B$ (26)</td>
<td>→ A$ (1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

And A$ are independent of each other.

A DEFM mode is canceled by declaring an array using the DIM statement or by DIM ALL.

**AMPLE:**

```plaintext
M A (m, n, o) ALL  
M ALL
```

Declaring 3-dimensional array A (m, n, o)

Only cancels DEFM mode.

```plaintext
DIM mode → DIM mode
M mode → DEFM mode
```

At this time, variables A( ) and A$ ( ) are erased.

This cannot be used within a FOR-NEXT loop.

<table>
<thead>
<tr>
<th>number of variables to be expanded</th>
</tr>
</thead>
</table>

### 2-2 Using DEFM Statement Arrays

When defining arrays using the DEFM statement, a variable area should be reserved for the variables used. Failure to do so will result in an OM error (memory over error) when DEFM is specified or when the program is executed. Should an OM error occur, the CLEAR statement must be used to reserve a variable area before the next DEFM specification. The following shows calculation of the required variable area size.

- **Variable:** 62 + 3 (bytes)
- **Z variables:** (8 + 4) × number of variables (bytes)
- **A$-Z$ variables:** (8 + 5) × number of variables (bytes)
- **Variable expansion:** 7 + 8 × number of variables to be expanded + 7 + 9 × number of variables to be expanded
8-2-3 DEFM Mode Displays

Using the DEFM specification in manual (direct key input) execution displays the number of variables in the DEFM array. This DEFM display does not appear when DEFM is specified within a program.

EXAMPLE:

```
D  E  F  M  24
A -- Z: 26  DEFM: 0
```

D  E  F  M  10  E16
A -- Z: 26  DEFM: 10

8-2-4 CLEAR Command, DIM Command, DEFM Command
In DIM Mode and DEFM Mode

Executing the CLEAR statement with this unit clears the contents of variables and reserves a variable area. Executing this command in each mode produces the following results.

<table>
<thead>
<tr>
<th>Operation</th>
<th>DIM Mode</th>
<th>DEFM Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delets variable contents</td>
<td>Delets variable contents</td>
</tr>
<tr>
<td></td>
<td>Deletes variable contents and reserves variable area</td>
<td>Deletes variable contents, cancels DEFM mode and reserves variable area</td>
</tr>
<tr>
<td></td>
<td>Enters DEFM mode and internally executes ERASE A, AS: DIM Ai( ), AS()</td>
<td>Displays number of arrays when executed manually</td>
</tr>
<tr>
<td></td>
<td>Executes DIM</td>
<td>Cancels DEFM and executes ERASE A, AS</td>
</tr>
</tbody>
</table>

* DEFM specification is not required for FX-790P and FX-730P programs which employ DIM statements.

8-3 LOADING PB-100 SERIES PROGRAMS

The following commands are executed in the BASIC mode to load PB-100 series programs from cassette tape.
PBLOAD : Program file
PBLOAD ALL : All files

Executing these commands loads the programs and automatically modifies them to allow execution on the FX-850P.

EXAMPLE:

PBLOAD from a cassette tape containing the following programs:

TEST 1 (memo file)
TEST 2 (data file)
TEST 3 (all files)
TEST 4 (program file)
8-4 READING PB-100 SERIES DATA

The following commands are available for reading of PB-100 series data files and DATA BANK files:
PBLOAD # (DATA BANK file)
PBGET (data file)

EXAMPLE:
Execute PBGET for filename TEST 5 containing the data S, A, B, C, D.

* Program, data, DATA BANK data, and all files saved to cassette tape using this unit cannot be read by PB-100 series computers.

8-5 COMMAND FORMAT

PBLOAD [ 0 ] ["filename"]

1. 0 : positive phase  1 : reverse phase (default = positive phase)
   Attempt using the phase which is opposite the current setting if problems are experienced during PBLOAD operations.
2. This command loads file under the specified filename into the current program area.
3. This command must be executed in the BASIC mode.
4. Execution of this command reads the specified program from tape and converts it from PB-100 series format to FX-850P/FX-880P format.
5. Execution of this command can be terminated at any time by pressing the ESC key.
PBLOAD ALL \[
\begin{bmatrix}
0 \\
1 \\
\end{bmatrix}
\] ["filename"]

1. This command loads all files under the specified filename into program areas P0 through P9.
Operation is identical to PBLOAD.

PBLOAD \# \[
\begin{bmatrix}
0 \\
1 \\
\end{bmatrix}
\] ["filename"] [, M]

1. This command reads the data under the specified filename into the DATA BANK.
2. The [, M] specification appends the read data to the end of the data currently stored in the DATA BANK.
Operation is identical to PBLOAD.

PBGET \[
\begin{bmatrix}
0 \\
1 \\
\end{bmatrix}
\] ["filename"]

1. This command reads the data file contents under the specified filename and assign then to S variable, A( ) array, or AS( ) array.
2. String data and numeric data are automatically assigned to the proper corresponding array.
3. This command can be executed in both the BASIC mode and CAL mode, but must be preceded by array declaration using the DIM statement or DEFM statement.
4. Data are automatically converted (Converting...displayed) before being assigned to variables.

NOTES
• Suspending execution of PBLOAD, PBLOAD ALL, PBLOAD #, and PBGET using the STOP key and then resuming operation should be avoided.
• A PO error (program error) or DA error (data error) is generated when program or data load is unsuccessful using a PB command.
• The following variables are used as work areas when any of the PB commands are executed:
  a, b, c, d, e, f, g, h, i, j, k, l, m, n,
  o, p, q, r, s, t, u, v, w, x, y
  aS, bS, cS, dS, eS, fS, gS, hS, iS, jS, kS, lS, mS, nS,
  oS, pS
• STAT, EOX, EOY, PUT and GET commands are converted to the code ??? when loaded to a PB series computer. Executing programs with this code generates an SN error (syntax error). See section 8-1 for details on manual conversion of code ???.
• Execution of a PB command while in the PRT mode (PRT symbol on display) automatically cancels the PRT mode.
9-1  FILING DEVICES

Besides execution of programs currently stored in memory, this unit can also employ cassette tapes for data and program storage. Data and programs can also be exchanged with other devices via a communications circuit. The OPEN, CLOSE, PRINT #, INPUT #, SAVE, and LOAD commands are used for these purposes.

- Device Names
When using the commands noted above for file interchanges, it is first necessary to specify whether cassette tape or the communications circuit is to be employed. This is known as specifying a "device name". The following table shows the available device names:

<table>
<thead>
<tr>
<th>DEVICE NAME</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAS0:</td>
<td>Cassette tape recorder (positive phase)</td>
</tr>
<tr>
<td>CAS1:</td>
<td>Cassette tape recorder (reverse phase)</td>
</tr>
<tr>
<td>COM0:</td>
<td>Communications circuit (RS-232C)</td>
</tr>
</tbody>
</table>

- Filenames
Once the device to be used is determined, the next thing to do is assign a "filename" to the file. A filename can be any combination of numeric and alphabetic characters up to eight characters in length.

The unit automatically disregards any filename input exceeding the first eight characters.

* Only one file can be open at any time. A program or subroutine, which successively opens and closes files as they are required, should be prepared when multiple files need to be accessed.

- File Descriptors
A "file descriptor" is actually a combination of a device name and filename. When using the communications circuit, the file descriptor also specifies various communications parameters in addition to the device name and filename.

9-2  FILE DESCRIPTORS

9-2-1  Cassette Tapes
The following shows the file descriptor when a cassette tape is used for file storage.

```
{ CAS0 : (S) filename }
{ CAS1 : (F) filename }
```
Phase specification

<table>
<thead>
<tr>
<th>CAS0 : Positive phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAS1 : Reverse phase</td>
</tr>
</tbody>
</table>

Speed specification

| S : 300 baud (300 bits/second) |
| F : 1200 baud (1200 bits/second) |

EXAMPLE 1:
Reading data of a file named TEST, at positive phase, 300 baud

    OPEN ""CAS0 : (S) TEST"" FOR INPUT AS #1

The file descriptor in this case is CAS0 : (S) TEST.

EXAMPLE 2:
Writing data to a file named SAMPLE, at positive phase, 1200 baud

    SAVE ""CAS0 : (F) SAMPLE"

The file descriptor in this case is CAS0 : (F) SAMPLE.

EXAMPLE 3:
Reading of a program under the filename AD1990

    LOAD "AD1990"

This is the same as LOAD ""CAS0 : (F) AD1990"

The file descriptor commands which can be used as outlined above are:

    OPEN, LOAD, SAVE, LOAD #, SAVE #, and VERIFY

* CAS0 : and CAS1 : specify the read phase (positive/reverse) of data recorded on tape. CAS0 : is generally used, but CAS1 : can be used for certain tape recorders.
* An attribute is automatically assigned when a program or data is written to cassette tape. Attributes can be checked when the tape is read.

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Binary file (program file)</td>
</tr>
<tr>
<td>A</td>
<td>All file (program file saved using SAVE ALL command)</td>
</tr>
<tr>
<td>S</td>
<td>Sequential file (data file, memo file, or program file saved in ASCII format)</td>
</tr>
</tbody>
</table>

9-2-2 Communications Circuit

The file descriptor for communications circuit data interchange is as follows:

    COM0 : [(Speed), [Parity], [Data], [Stop], [CS], [DS], [CD], [Busy], [Code]]

1. Speed (baud rate : data transfer speed)
   
   6 : 4800 baud (4800 bits/sec)
   5 : 2400 baud (2400 bits/sec)
   4 : 1200 baud (1200 bits/sec)
   3 : 600 baud (600 bits/sec)
   2 : 300 baud (300 bits/sec)
   1 : 150 baud (150 bits/sec)

NOTE: 2400 (5) or less should be specified when the communications circuit is specified using the OPEN statement.
Parity (parity bit : check bit for data send)
N : No parity bit
E : Even parity
O : Odd parity

Data (data bit : number of bits representing one character)
7 : JIS 7 bit (7 bits/character)
8 : JIS 8 bit (8 bits/character)

Stop (stop bit : bit(s) at end of a character signifying character end)
1 : 1 bit
2 : 2 bits

CS (Clear to send (CTS) : control function to inform partner device whether or not data can be sent)
C : Used — The sending device waits until CS is ON.
N : Not used

DS (Data set ready (DSR) : function to inform normal operation of partner device)
D : Used — An NR error is generated when data are received while DS is OFF. When sending, the device waits until DS is ON.
N : Not used

CD (Carrier detect (CD) : function to inform partner device ready to receive data)
C : Used — An NR error is generated when data are received while CD is OFF.
N : Not used

Busy (Busy, (XON/OFF), function to temporarily suspend data send)
B : Used — A send suspend request is sent to the partner device when the remaining number of characters to be read at one time is less than 64 during data receive. During data send, send is temporarily suspended when a send suspend request is received from the partner device. Sending is resumed upon a send start request.
N : Not used

Code (Input/Output code system (SI/SO) : Used to send character expressed as eight bits when data bits specified as seven)
S : Used — This function is only applicable when the data bit parameter (above 3) is specified as seven bits. The SO code (0Eh) is sent before codes 80h or higher in this case to enter the SO mode. Codes 7Fh or lower are sent preceded by the SI code (0Fh) to enter the SI mode.
The S specification should be used whenever it is necessary to send data with codes 80h or higher as 7 bit data.
N : Not used

Parameter Default Values
COM0 : 2, E, 8, 1, N, N, N, B, N
Baud rate : 300 baud (300 bits/sec)
Parity : Even
Data bits : 8 bits
Stop bit : 1 bit
CS : Not used
DS : Not used
CD : Not used
Busy : XON/XOFF
Code : SI/SO – Not used
EXAMPLE 1:
Send the character string noted below to RS-232C using the parameters listed.

Baud rate : 300 baud → 2
Parity : Even → E
Data bits : 8 bits → 8
Stop bit : 1 bit → 1
CS : Not used → N
DS : Not used → N
CD : Not used → N
Busy : Used → B
Shift in/out : Not used → N

10 OPEN "COM0: 2, E, 8, 1, N, N, B, N" AS #1
20 PRINT #1, "HELLO."
30 CLOSE

EXAMPLE 2:
Receive character string from RS-232C as above.

10 OPEN "COM0: 2, E, 8, 1, N, N, N, B, N" AS #1
20 INPUT #1, AS
30 CLOSE

* Communications are performed via the RS-232C terminal.
* Data interchange is performed using a full-duplex (both sides can communicate simultaneously, as with a standard telephone), start-stop system (typical computer communications system, also known as asynchronous).
FORMAT ELEMENTS

The method for entering statements is explained below.
- Words in bold type are commands or functions, and they must be entered as shown.
- Braces indicate that one of the parameters enclosed must be specified.
- Commas contained in braces must be written in the position shown.
- Brackets indicate that the parameters enclosed may be omitted. Brackets themselves are not entered.
- An asterisk indicates that the term preceding it may appear more than once.
- Numeric expressions—Constants, expressions, and numeric variables (e.g. 10, 10 + 25, A, unit cost * quantity)
- String expressions—String constants, string variables, and string expressions (e.g. "ABC", A$, and A$ + BS)
- Expressions—General term for numeric and string expressions
- Arguments—Elements used by commands and functions
- P........Can only be executed in a program.
- M........Can only be executed manually.
- A........Can be executed both manually and in a program.
- F........Function instruction that can be executed both manually and in a program.

Example: MIDS function

MIDS ( string array , position [ , number of characters ] )

String expression Numeric expression Numeric expression

The term "string expression" under "string array" describes that array. Likewise, "numeric expression" under "position" and "numeric expression" under "number of characters" are descriptors. Also, since the comma and number of characters are enclosed in brackets, they may be omitted.

Example: GOSUB Statement

GOSUB { branch line number }

Line number

# program area number Single character; 0–9

This example illustrates two descriptors for GOSUB: the line number of the subroutine to which the program branches and filename to which the program branches.
PASS

PURPOSE: Specifies or cancels a password.

FORMAT:  

PASS "password"

String expression

EXAMPLE:  

PASS "TEXT"

PARAMETERS:  
1. Registering a single password makes it the password for all program areas (P0 ~ P9) and for DATA BANK function.
2. The password must be a string of 1 ~ 8 characters.
3. All characters after the first 8 are ignored when 9 or more characters are entered.

EXPLANATION:
1. The password is used to protect programs and DATA BANK data.
2. The password can be registered in both the CAL mode and BASIC mode.
3. Executing this command registers a password when no password previously exists.
4. Executing the PASS statement using a previously registered password cancels the password. Specifying a password that is different from that registered, results in a PR error.
5. The following operations and commands cannot be executed when a password is registered:
   - Program write
   - MEMO IN mode specification
   - MEMO search
   - LIST, LLIST, LIST ALL, LLIST ALL, LIST#, LLIST#, NEW, NEW ALL, NEW#
   - EDIT
   - SAVE, SAVE# to RS-232C
   - SAVE, SAVE# to cassette tape in ASCII format
6. Executing SAVE and SAVE ALL to cassette tape applies the password to the saved program.
7. Loading a program (using LOAD or LOAD ALL) which is protected by a password into the computer causes the password of the loaded program to be registered as the computer password. A PR error is generated when the current password differs from the password of the loaded program.
NEW [ALL]

**PURPOSE:** Deletes a program.

**FORMAT:** NEW [ALL]

**EXAMPLE:** NEW

**EXPLANATION:**
1. Deletes the program in the currently specified program area when ALL is omitted.
   Variables are not cleared.
2. "Ready Pn" is displayed on the screen after the program is deleted, and the computer stands by for command input.
3. All files that are currently opened are closed.
4. This command cannot be executed for program files that are protected by a password.
5. Attempting to use this command in the CAL mode results in an FC error.
6. Specifying NEW ALL clears the programs in all program areas and all variables.
7. This command cannot be included within a program.

CLEAR

**PURPOSE:** Clears all variables and determines the variable area size in accordance with the parameter entered. Also closes any files that are open.

**FORMAT:** CLEAR [variable area size]

**EXAMPLE:** CLEAR 400

**PARAMETERS:** variable area size: Numeric expression
Determines the areas used for variables. The initial setting when ALL RESET is executed depends upon total memory capacity.

<table>
<thead>
<tr>
<th>MEMORY CAPACITY</th>
<th>VARIABLE AREA SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less than 32KB</td>
<td>1536 bytes</td>
</tr>
<tr>
<td>32KB and over</td>
<td>8192 bytes</td>
</tr>
</tbody>
</table>

**EXPLANATION:**
1. Clears all variables.
2. Closes all open files and clears the FOR–NEXT and GOSUB stack.
3. Variable area cannot be set during program execution.

**SEE:** FRE
MEMORY MAP

<table>
<thead>
<tr>
<th>Address</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>System area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAM area for stack</td>
<td>768 bytes</td>
</tr>
<tr>
<td></td>
<td>Character variable data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>↓ Variable free area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>↑ (capacity can be referenced using FRE0)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Numeric variable data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Variable table</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P0 area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P1 area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P9 area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DATA BANK area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>↓ Free area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(capacity can be referenced using FRE1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program control area</td>
<td></td>
</tr>
</tbody>
</table>

EXPANDED MEMORY CONFIGURATION (UNIT = BYTES)

<table>
<thead>
<tr>
<th></th>
<th>FX-850P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>FRE 1</td>
</tr>
<tr>
<td></td>
<td>FRE 2</td>
</tr>
<tr>
<td></td>
<td>User’s area</td>
</tr>
<tr>
<td>RP-8 RAM expansion pack</td>
<td>FRE 1</td>
</tr>
<tr>
<td></td>
<td>FRE 2</td>
</tr>
<tr>
<td></td>
<td>User’s area</td>
</tr>
<tr>
<td>RP-33 RAM expansion pack</td>
<td>FRE 1</td>
</tr>
<tr>
<td></td>
<td>FRE 2</td>
</tr>
<tr>
<td></td>
<td>User’s area</td>
</tr>
</tbody>
</table>

FRE 1 capacity value is when no programs or DATA BANK data are stored. FRE 2 capacity can be changed using the CLEAR command.
STACK AREA

- I/O BUFFER
  - For file opened using OPEN statement
  - Capacity: 296 bytes (cassette tape) or 42 bytes (RS-232C)

- CHARACTER OPERATION STACK
  - For character operations

- STACK FREE AREA
  - For confirmation of operation and array variable contents

- DATA STACK
  - For FOR loop execution
  - Capacity: 26 bytes/loop

- FOR STACK
  - For FOR loop execution
  - Capacity: 26 bytes/loop

- GOSUB STACK
  - For GOSUB branching
  - Capacity: 8 bytes/branch

FRE

Purpose: Returns memory area size in accordance with argument.

Format: FRE( argument Numeric expression )

Example: PRINT FRE 0

Parameters: argument: Integer in the range of 0 ≤ argument < 3

Explanation:
1. parameter = 0: Returns unused memory in variable area in byte units
2. parameter = 1: Returns unused memory in program or in DATA BANK area in byte units
3. parameter = 2: Returns overall variable area in byte units

See: CLEAR
LIST [ALL]

PURPOSE: Displays all or a part of the currently specified program.

FORMAT:

\[
\text{LIST} \left\{ \begin{array}{c}
\text{start line number} \hspace{1cm} \text{end line number} \\
\text{Line number} \hspace{1cm} \text{Line number}
\end{array} \right. \\
\text{[.]} \\
\text{[ALL]}
\]

EXAMPLE:

LIST 100
LIST 100 - 300
LIST - 400
LIST

PARAMETERS:
1. start line number: Integer in the range of \(1 \leq \text{line number} \leq 65535\) (first line number when omitted)
2. end line number: Integer in the range of \(1 \leq \text{line number} \leq 65535\) (end line number when omitted)

EXPLANATION:
1. Displays the currently specified program in the range specified by the line numbers.
2. A minus sign must be used as the delimiter between line numbers.
3. The following five examples illustrate specification of the display range.
   a) LIST \(\text{ALL}\) (All lines from beginning of program)
   b) LIST 30 \(\text{30}\) (Line 30)
   c) LIST 50 - 100 \(\text{50 - 100}\) (Lines 50 through 100)
   d) LIST 200 - \(\text{200 -}\) (From line 200 through end of program)
   e) LIST - 80 \(\text{- 80}\) (From beginning of program through line 80)
4. Using a period in place of the line number displays the most recently handled (i.e. written, edited, executed). If a program is halted during execution by an error, executing "LIST ." displays the line in which the error was generated.
5. When the specified start line number does not exist, the first line number above that specified is taken as the start line number.
6. When the specified end line number does not exist, the greatest line number not exceeding that specified is taken as the end line number.
7. The start line number must be smaller than the end line number.
8. LIST command execution can be halted by pressing the \(\text{STOP}\) key.
9. Press the \(\text{STOP}\) key to momentarily halt LIST command execution. To restart execution, press the \(\text{RESET}\) key or one of the alphanumeric keys.
10. The computer stands by for command input after the program list is displayed.
11. This command cannot be used when a password is registered.
12. This command cannot be used in the CAL mode.
13. Specifying ALL displays all programs in sequence from area P0 through P9.

SEE: EDIT, VARLIST, LLIST
EDIT

PURPOSE: Enters the BASIC edit mode.

FORMAT: 
EDIT [start line number]

EXAMPLE: EDIT 100

PARAMETERS: start line number: Integer in the range of 1 ≤ line number ≤ 65535 (first line number when omitted)

EXPLANATION:
1. Enters the BASIC edit mode and displays the program from the specified line number.
   The cursor is displayed and editing becomes possible when either the ≡ or ⊖ key is pressed.
2. Using a period in place of the line number displays the most recently handled (i.e. written, edited, executed). If a program is halted during execution by an error, executing "EDIT ." displays the line in which the error was generated.
3. When the specified start line number does not exist, the first line number above that specified is taken as the start line number.
4. This command cannot be used when a password is registered.
5. This command cannot be used in the CAL mode.
6. This mode is canceled by pressing the ≡ key.

SEE: LIST, LLIST

VARLIST

PURPOSE: Displays variable names and array names.

EXAMPLE: VARLIST

EXPLANATION:
1. Displays all currently existing variable names and array names.
2. Press the ≡ key to momentarily halt VARLIST command execution. To restart execution, press the ≡ key or one of the alphanumeric keys.

SAMPLE EXECUTION: VARLIST ≡

A AB R ACS( )

This command displays all variable names and array names currently stored in memory.
PURPOSE: Executes a program.

FORMAT: \[ \text{RUN} \ [ \text{execution start line} ] \]
\[ \text{Line number} \]

EXAMPLE: \[ \text{RUN} \]
\[ \text{RUN 100} \]

PARAMETERS: start line number: Integer in the range of \( 1 \leq \text{line number} \leq 65535 \)

EXPLANATION:
1. Execution starts from the beginning of the program when the line number is omitted.
2. When the specified start line number does not exist, the first line number above that specified is taken as the start line number.
3. This command closes all files that are open.
4. Variable and array values are not cleared.
5. This command cannot be used within a program.
6. This command cannot be used in the CAL mode.

SAMPLE PROGRAM:

\[ \text{RUN 100} \]

Executes program from line number 100.
**TRON**

**PURPOSE:** Specifies the trace mode.

**EXAMPLE:** TRON

**EXPLANATION:**
1. Switches the trace mode ON and TR appears on the display.
2. All subsequent program execution is accompanied by a display of the area name and line number. The first two lines are displayed, and execution is suspended.
   Program execution can be resumed at this time by pressing EX.
3. The program stays in the TRON mode until the TROFF statement is executed or the power is switched OFF.

**SEE:** TROFF

**SAMPLE EXECUTION:**

```
TRON EX
RUN EX
```

P0 represents currently specified program area and 10 currently executed line number.

---

**TROFF**

**PURPOSE:** Cancels the trace mode.

**EXAMPLE:** TROFF

**EXPLANATION:**
Cancels the trace mode (entered using the TRON statement).

**SEE:** TRON
Purpose: Terminates program execution.

Example: END

Explanation:
1. Terminates program execution, and the computer stands by for command input.
2. Closes all files that are open.
3. Variables and arrays are not cleared.
4. Any number of END statements can be used in a single program. Program execution is terminated and open files are closed automatically at the end of the program even if an END statement is not included.

Sample Program:

```
10 FOR I = 1 TO 20
20 IF I > 10 THEN END
30 PRINT I ;
40 NEXT I
```

Displays values of I in FOR–NEXT loop.
Program ends when I exceeds 10.
STOP

PURPOSE: Temporarily halts program execution.

EXAMPLE: STOP

EXPLANATION:
1. Temporarily halts program execution and STOP appears on the display. Execution can be resumed by pressing EXIT.
2. Pressing EXIT while execution is halted by the STOP command displays the current program area and line number.
3. Such commands as PRINT can be executed while execution is halted by the STOP command. Manual calculations can also be performed in the CAL mode.
4. Open files, variable values and array values are retained as they are at the point when execution is halted.
5. The STOP status is canceled when an error is generated, the mode is changed, or the program is edited while program execution is halted.

SAMPLE PROGRAM:

    10 FOR I = 1 TO 10
    20 IF I = 6 THEN STOP : PRINT
    30 PRINT I;
    40 NEXT I

Displays values of I in FOR – NEXT loop.
Execution is halted when I equals 6. Next, pressing EXIT resumes execution.
GOTO

Purpose: Branches unconditionally to a specified branch destination.

Format: 

\[
\text{GOTO \{ \\
\quad \text{branch destination line number} \\
\quad \text{Line number} \\
\quad \# \text{program area number} \\
\quad \text{Single character; 0~9}
\}}
\]

Sample: 

\begin{align*}
\text{GOTO} & \quad 1000 \\
\text{GOTO} & \quad \#7
\end{align*}

Parameters: 

1. Branch destination line number: Integer in the range of \(1 \leq \text{line number} \leq 65535\)
2. Program area number: Single character, 0~9

Explanation:

1. Specifying a line number causes program execution to jump to that line number in the current program area.
2. Specifying a program area number causes program execution to jump to the first line number of the specified program area.
3. A UL error is generated when the specified line number does not exist.

Sample Program:

\begin{align*}
10 & \quad \text{PRINT ""PRESS [BRK]"" ;} \\
20 & \quad \text{PRINT ""TO HALT EXECUTION"" ;} \\
30 & \quad \text{GOTO 10}
\end{align*}

Line 30 returns execution to line 10.
This loop continues until the \# key is pressed.
GOSUB

PURPOSE: Jumps to a specified subroutine.

FORMAT: \[
\begin{align*}
\text{Line number} & \quad \text{branch destination line number} \\
\text{GOSUB} & \quad \# \text{program area number} \\
& \quad \text{Single character; 0–9}
\end{align*}
\]

EXAMPLE: GOSUB 100
          GOSUB #6

PARAMETERS: 1. branch destination line number: Integer in the range of 1 ≤ line number ≤ 65535
              2. program area number: Single character, 0–9

EXPLANATION:
- Program execution branches to the subroutine that starts at the specified line number.
- Execution is returned from the subroutine by the RETURN statement.
- Subroutines can be nested up to 96 levels. Exceeding this value results in an OM error.
- A UL error is generated when the specified line number does not exist.
- CLEAR command cannot be used within a subroutine.

SAMPLE PROGRAM:

10 REM***MAIN***
20 GOSUB 40
30 END
40 REM***SUBROUTINE 1***
50 PRINT "SUBROUTINE 1";
60 GOSUB 80
70 RETURN
80 REM***SUBROUTINE 2***
90 PRINT "SUBROUTINE 2"
100 RETURN

Line 20 branches to subroutine beginning at line 40, and line 60 branches to subroutine beginning at line 80.
**RETURN**

**PURPOSE:** Returns execution from a subroutine to the main program.

**FORMAT:** RETURN

**EXAMPLE:** RETURN

**EXPLANATION:**
1. Returns program execution to the statement immediately following the statement which originally called a subroutine.
2. A GS error is generated when the RETURN statement is executed without first executing a GOSUB statement.

**SEE:** GOSUB, ON~GOSUB

**SAMPLE PROGRAM:**

```
10 REM SUBROUTINE
20 GOSUB 100
30 END
100 PRINT "SUBROUTINE 1"
110 GOSUB 200
120 RETURN
200 PRINT "SUBROUTINE 2"
210 RETURN
```

RETURN in line 120 returns to line 20, while line 210 returns to line 110.
ON GOTO

PURPOSE: Jumps to a specified branch destination in accordance with a specified branching condition.

Syntax: ON <condition> GOTO [branch destination] [ . [branch destination]]*

Branch destination: {destination line number
Line number
# program area number
Single character; 0–9

EXAMPLE: ON A GOTO 100, 200, 300

PARAMETERS:
1. branch condition: Numeric expression truncated to an integer
2. line number: Integer in the range of 1 ≤ line number ≤ 65535
3. program area number: Single character, 0–9

EXAMPLE PROGRAM

10 INPUT "1 OR 2" ; A
20 ON A GOTO 40, 50
30 END
40 PRINT "ONE" : END
50 PRINT "TWO"

Execution jumps to line 40 if 1 is entered or to line 50 if 2 is entered. Otherwise, execution terminates at line 30.
ON GOSUB

PURPOSE: Jumps to a specified subroutine in accordance with a specified branching condition.

FORMAT: ON \( \text{condition} \) GOSUB \( [ \text{branch \ destination} ] \) \( \text{[branch \ destination]} \)*

Branch destination:
\[
\begin{align*}
\text{destination branch line number} \\
\text{Line number} \\
\# \text{program area number} \\
\text{Single character; } 0-9
\end{align*}
\]

EXAMPLE: ON A GOSUB 1000, 1100, 1200

PARAMETERS: 1. branch condition: Numeric expression truncated to an integer
2. line number: Integer in the range of \( 1 \leq \text{line number} \leq 65535 \)
3. program area number: Single character, 0–9

EXPLANATION:
1. The GOSUB statement is executed in accordance with the value of the expression used for the branch condition. For example, execution jumps to the first branch destination specified when the value is 1, to the second destination when the value is 2, etc.
2. Program execution does not branch and execution proceeds to the next statement when the value of the branch condition is less than 1, or if a branch destination corresponding to that value does not exist.
3. Up to 99 branch destinations may be specified.

SEE: RETURN

SAMPLE PROGRAM:

\begin{verbatim}
10   S1 = 0 : S2 = 0
20   FOR I = 1 TO 100
30   ON (I MOD 2) + 1 GOSUB 1000, 1100
40   NEXT I
50   PRINT "S1 =" ; S1
60   PRINT "S2 =" ; S2
70   END
1000  S1 = S1 + I : RETURN
1100  S2 = S2 + I : RETURN
\end{verbatim}

S1 calculates sum of even numbers from 1 to 100, S2 calculates sum of odd numbers from 1 to 100.
F ~ THEN ~ ELSE/IF ~ GOTO ~ ELSE

REPOSE: Executes the THEN statement or GOTO statement when the specified condition is met. The ELSE statement is executed when the specified condition is not met.


Branch destination: destination branch line number Line number
# program area number Single character; 0 ~ 9

AMPLE: IF A = 0 THEN 300 ELSE 400
IF KS = "Y" THEN PRINT X ELSE PRINT Y

RAMETERS: 1. branch condition: Numeric expression truncated to an integer
2. line number: Integer in the range of 1 ≤ line number ≤ 65535
3. program area number: Single character, 0 ~ 9

PLANATION:
The statement following the THEN clause is executed, or execution jumps to the destination specified by the GOTO statement when the branch condition is met.
If the branch condition is not met, the statement following the ELSE statement is executed, or the program jumps to the specified branch destination. Execution proceeds to the next program line when the ELSE statement is omitted.
The format "IF A THEN - " results in the condition being met when value of the expression (A) is not 0 (absolute value of A < 1 × 10^-99). The condition is not met when the value of the expression is 0.
F statements can be nested (an IF statement may contain other IF statements). In this case, the THEN ~ ELSE statements are related by their proximity. The GOTO ~ ELSE combinations have the same relationships.
F ~ THEN IF THEN ~ ELSE IF ~ THEN ~ ELSE ~ ELSE ~

AMPLE PROGRAM:

10 INPUT "'1 TO 9'" ; A
20 IF (0<A) AND (A<10) THEN PRINT "'GOOD!'" ELSE 10

"GOOD" is displayed for input values from 1 to 9. Re-input is requested for other values.
FOR ~ NEXT

PURPOSE: Executes the program lines between the FOR statement and NEXT statement and increments the control variable, starting with the initial value. Execution is terminated when value of the control variable exceeds the specified final value.

FORMAT: FOR control variable name = initial value
          Numeric expression
          TO final value [STEP increment]
          Numeric expression Numeric expression
          NEXT [Control variable name] [, Control variable name]*

EXAMPLE: FOR I = 1 TO 10 STEP 0.1
          
          NEXT I

PARAMETERS: 1. control variable name: Array variables cannot be used.
              2. initial value: Numeric expression
              3. final value: Numeric expression
              4. increment: Numeric expression (default value = 1)

EXPLANATION:
1. None of the statements between FOR and NEXT are executed and the program proceeds to the next executable statement after NEXT when the initial value is greater than the final value.
2. Each FOR requires a corresponding NEXT.
3. FOR ~ NEXT loops can be nested (a-FOR ~ NEXT loop can be placed inside another FOR ~ NEXT loop). Nested loops must be structured as shown below with NEXT appearing in reverse sequence of the FOR (e.g. FOR A, FOR B, FOR C ~ NEXT C, NEXT B, NEXT A).

```
10 FOR I = 1 TO 12 STEP 3
20 FOR J = 1 TO 4 STEP 0.5
30 PRINT I, J
40 NEXT J
50 NEXT I
60 END
```
4. FOR ~ NEXT loops can be nested up to 29 levels.
5. The control variable may be omitted from NEXT. However, use of the control variable in the NEXT statement is recommended when using nested loops.
NEXT statements can be chained by including them under one NEXT statement, separated by commas.

```
10 FOR I = 1 TO 12 STEP 3
20 FOR J = 1 TO 4 STEP 0.5
30 PRINT I, J
40 NEXT J
50 NEXT I
60 END
```

The control variable retains the value which exceeds the final value (and terminates the loop) when loop execution is complete. With the loop FOR I = 3 TO 10 STEP 3, for example, the value of control variable I is 12 when execution of the loop is complete.

Jumping out of a FOR – NEXT loop is also possible. In this case, the current control variable value is retained in memory, and the loop can be resumed by returning with a GOTO statement.
REM( ' )

PURPOSE: Allows remarks or comments to be included within a program. This command is not executed.

FORMAT: 

```
REM
  comments
```

String expression

EXAMPLE: REM or '

PARAMETERS: 

PARAMETERS: comments: String expression (internal codes 20 to 7E and 80 to FB)

EXPLANATION:

1. Including an apostrophe or REM statement following the line number indicates that the following text is comments and should be ignored in program execution.
2. The apostrophe may be included at the end of any executable statement to indicate that the following text is comments. The REM statement can only be used at the beginning of a line.
3. Any command following the REM statement is treated as a comment and is not executed.

```
PRINT A: REM 123 123 is treated as a comment.
     Comments
PRINT A REM 123 SN error occurs.
PRINT A ' 123 123 is treated as a comment.
     Comments
```

4. An apostrophe is entered by pressing the key following the key.

SAMPLE PROGRAM:

```
10 ' REM( ' ) indicates comment
```
PURPOSE: Assigns the value of an expression on the right side of an equation to the variable on the left side.

FORMAT: [LET] numeric variable name = Numeric expression.
[LET] string variable name = String expression

EXAMPLE: LET A = 15
LET K$ = "123"

EXPLANATION:
1. Assigns the value of an expression on the right side of an equation to the variable on the left side.
2. Numeric expressions can only be assigned to numeric variables, and string expressions can only be assigned to string variables. A TM error is generated when an attempt is made to assign a string expression to a numeric variable, and vice versa.
3. LET may be omitted.

AMPLE PROGRAM:

10 LET A = 10
20 B = 20
30 PRINT A ; B

Assigns 10 to variable A and 20 to variable B, and displays both.
DATA

PURPOSE: Holds data for reading by the READ statement.

FORMAT: DATA [ data ] [ , [ data ]]*
        Constant    Constant

EXAMPLE: DATA 10, 5, 8, 3
        DATA CAT, DOG, LION

PARAMETERS: 1. data: String constants or numeric constants
             2. string constants: Quotation marks are not required unless the string
                                contains a comma which is part of the data. A null data string (length
                                0) is assumed when data is omitted from this statement.

EXPLANATION:
1. This statement can be used anywhere in the program to hold data to be read by the READ
   command.
2. Multiple data items are separated by commas.

SEE: READ, RESTORE

SAMPLE PROGRAM:

   10 READ A$
   20 RESTORE 60
   30 READ B$
   40 PRINT A$ + " " + B$
   50 DATA AD 1990, ABC
   60 DATA DEFG

Character data "AD1990" and "DEFG" read from lines 50 and 60, and displayed.
RPOSE: Reads the contents of the DATA statement into memory.

RMAT: READ Variable name [,Variable name ]*

AMPLE: READ A, B
       READ C$, X, Y

RAMETERS: Variable name

PLANATION:
Assigns the data contained in a DATA statement to the variables on a one-by-one basis. Numeric data can only be assigned to numeric variables, and string data can only be assigned to string variables. A TM error is generated when an attempt is made to assign string data to a numeric variable, and vice versa.
The data in DATA statements is read from the lowest line number in ascending order. Data are read in order from the beginning of a DATA statement. The first execution of the READ statement reads the first data item contained in the first DATA statement. Subsequent executions read data items in sequential order. The data line to be read can be specified using the RESTORE statement.

EE: DATA, RESTORE

AMPLE PROGRAM:

10 READ X
20 IF X > 0 THEN PRINT X ; : GOTO 10
30 END
100 DATA 1, 2, 3, 4, 5, 6, 7, 8, 9
110 DATA 9, 8, 7, 6, 5, 4, 3, 2, 1
120 DATA 0

Sequentially reads data beginning at line 100 and stops execution when 0 is encountered as data.
RESTORE

PURPOSE: Specifies a DATA line for reading by the READ statement.

FORMAT: 

    RESTORE \[ line number \]
    Numeric expression

EXAMPLES:

    RESTORE
    RESTORE 1000
    RESTORE (10*10)

    \[ \text{line } 100 \]

PARAMETERS: line number: Integer in the range of \( 1 \leq \text{line number} \leq 65535 \)

EXPLANATION:
1. The first DATA line in the program file containing the READ statement is the default option when the line number is omitted.
2. When a line number is specified, the first data item in the specified DATA line is read by the next READ statement execution. A UL error is generated when the specified line number does not exist, while a DA error is generated when no data exist in the specified line.
3. A numeric expression can be used for line number specification. In this case, the numeric expression must be enclosed in parentheses.

SEE: READ, DATA

SAMPLE PROGRAM:

```
10  READ X
20  IF X<>0 THEN PRINT X ;: GOTO 10
30  RESTORE 110
40  READ X
50  IF X<>0 THEN PRINT X ;: GOTO 40
60  END
100 DATA 1, 2, 3, 4, 5, 6, 7, 8, 9
110 DATA 9, 8, 7, 6, 5, 4, 3, 2, 1
120 DATA 0
```

Lines 10 – 20 read data from lines 100 – 120, while lines 30 – 50 read data from lines 110 – 120 then display them.
PRINT

PURPOSE: Displays data on the screen.

FORMAT: PRINT [output data] [output data] [output data]...

Output data: TAB (Tab specification), numeric expression, string array

EXAMPLE: PRINT "AD1990"

PARAMETERS: output data: Output control function, numeric expression, or string expression

EXPLANATION:
Output of a numeric or string expression displays the value or string on the screen. Control function output results in the operation determined by the function being performed. Numeric expressions are displayed in decimal notation with values longer than 10 digits.

a) Integers: Values less than \( 1 \times 10^3 \)

b) Fraction: Decimal fractions smaller than 10 digits

c) Exponent: Other values

A space is added after displayed numeric expressions, with negative expressions preceded by a minus sign, and positive expressions preceded by a space. Expressions are displayed as integers, fractions, or exponential expressions, with the display format automatically selected according to the value of the expression.

String expressions are displayed unchanged. There are, however, special operations for internal codes \( 00H - 1FH, 7FH \) (see CHARACTER CODE TABLE on page 395).

Output is displayed on the screen from the current position of the cursor to the right. A line feed results when the cursor reaches the last column on the last line of the screen (lower right), scrolling the entire screen upwards. Subsequent output is displayed from the beginning of the bottom line of the screen (lower left).

Separating output data with commas causes execution to be halted with each display (STOP appears on display). Pressing \( \text{Ex} \) executes a carrier return/line feed and proceeds to the next display.

Separating output data with semicolons causes each output to be displayed immediately following the previous output.

Including a semicolon at the end of this statement causes the cursor to remain at position immediately following the displayed output.

Ending this command with output data or a comma, causes execution to be halted following display of the output data (STOP appears on display). Pressing \( \text{Ex} \) executes a line change and proceeds to the next display.
9. Omitting the output data (PRINT command only) executes a line change without halting execution.
10. Execution is not halted when this statement is executed while in the print mode (TAB).
11. Execution is not halted when this statement is executed while in the manual mode.

SEE: TAB

SAMPLE PROGRAM:

```
10 PRINT "PRINT DISPLAYS MESSAGES"
20 PRINT "ON THE SCREEN"
```

PRINT statement displays message on screen.
POSE: Outputs a horizontal tab specification to the screen or printer.

MAT: TAB (tab specification)

Numeric constant or numeric variable

SAMPLE: PRINT TAB (5); "ABC"

PARAMETERS: tab specification: Numeric expression truncated to an integer in the range of 0 ≤ tab specification < 256.

PLANATION: Used in the PRINT, LPRINT, and PRINT# statements to specify a display position on a line. Spaces are inserted from the left end of the line to the specified position. The display position is determined by counting from the left end of the line (position 0) and to the right, up to the specified value.

A tab specification value which is less than the current printhead position causes the tabulation to be performed following a carrier return/line feed.

EXAMPLE PROGRAM:

10 FOR I = 0 TO 25
20 PRINT TAB (I); "ABCDEFG" ;
30 NEXT I

Prints successive lines of "ABCDEFG", with each line proceeding to the right.
LOCATE

PURPOSE: Moves the cursor to a specified position on the virtual screen.

FORMAT: LOCATE X-coordinate , Y-coordinate

EXAMPLE: LOCATE 10, 0

PARAMETERS:
1. X-coordinate: Numeric expression truncated to an integer in the range of 0 ≤ X-coordinate < 32
2. Y-coordinate: Numeric expression truncated to an integer in the range of 0 ≤ Y-coordinate < 8

EXPLANATION:
1. Locates the cursor at a specified position on the virtual screen.
2. The origin of the coordinates is the upper left corner of the screen (0, 0). The X coordinate value is incremented for each character position to the right. The Y value coordinate is incremented form each line down.

\[(0, 0) \rightarrow \circ \circ \rightarrow (31, 0)\]

\[(0, 7) \rightarrow \circ \circ \rightarrow (31, 7)\]

SAMPLE PROGRAM:

10  CLS
20  LOCATE 0, 0
30  PRINT "SCREEN UPPER LEFT" ;
40  GOTO 20

Displays message from upper left of display.
CLS

PURPOSE: Clears the display screen.

SAMPLE: CLS

EXPLANATION: The screen is cleared and the cursor is located at the home position. Pressing the END key or executing PRINT CHR$(12) produces the same result.

SAMPLE PROGRAM:

10 REM CLEAR SCREEN
20 CLS

Clears screen.
 PURPOSE: Specifies output format of numeric data.

 FORMAT: 

 \[
 \begin{align*}
 \text{F} & \quad \text{number of digits} \\
 & \quad \text{Single character; } 0-9 \\
 \text{SET} & \\
 \text{E} & \quad \text{number of digits} \\
 & \quad \text{Single character; } 0-9 \\
 \text{N} & 
\end{align*}
\]

 EXAMPLE: SET F3

 PARAMETERS: 

 - F \quad \text{number of digits} \\
   \quad \text{Single character; } 0-9 \\
   \text{Specifies number of decimal places.}

 - E \quad \text{number of digits} \\
   \quad \text{Single character; } 0-9 \\
   \text{Specifies number of significant digits.}

 - N \\
   \text{Cancels current specification.}

 EXPLANATION:

 1. This command specifies the number of decimal places and the number of significant digits for numeric data output to the display, printer, tape recorder, or RS-232C terminal.
 2. The number of decimal places can be specified within the range of 0 through 9.
 3. The number of significant digits can be specified within the range of 1 through 10. The statement SET E0 specifies the number of significant digits as 10.
 4. SET N cancels both specifications.
 5. Output values are rounded to the specified decimal places or to the specified significant digits.
 6. This command is only valid for output data. The mantissa part for internal calculations is still 12 digits.

 SAMPLE PROGRAM:

```plaintext
10   A = 10/3
20   SET F2
30   PRINT A
40   SET E2
50   PRINT A
60   END
```

<table>
<thead>
<tr>
<th>RUN</th>
<th>NEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.33</td>
<td></td>
</tr>
<tr>
<td>3.3E+00</td>
<td></td>
</tr>
<tr>
<td>3.3E+00</td>
<td></td>
</tr>
<tr>
<td>3.33333333</td>
<td></td>
</tr>
<tr>
<td>3.3333333</td>
<td></td>
</tr>
</tbody>
</table>

Result of 10/3 displayed with 2 decimal places, 2 significant digits, and specification canceled.
BEEP

PURPOSE: Sounds the buzzer.

FORMAT: \[
\text{BEEP} \left[ \begin{array}{c}
0 \\
1 \\
\text{Numeric expression}
\end{array} \right]
\]

EXAMPLE: BEEP 1

EXPLANATION:
1. A low tone is specified by BEEP or BEEP 0.
2. A high tone is specified by BEEP 1.
3. Numeric expressions can be in place of 0 and 1.

SAMPLE PROGRAM:

10 BEEP 1 : BEEP 0 : BEEP 1 : BEEP 0
INPUT

PURPOSE: Assigns keyboard data input to a variable.

FORMAT: INPUT ['"message" [ ; ] variable name [ , ["message" [ ; ] variable name ] ] ]

EXAMPLE: INPUT "YEAR = ", Y, "MONTH = ", M, "DAY = ", D

PARAMETERS:
1. message: Character string beginning with a string constant
2. variable name: Numeric variable name or string variable name

EXPLANATION:
1. Data can be input to the specified variable from the keyboard.
2. Messages included in the INPUT statement are displayed. A question mark is displayed following the message when a semicolon is included following the message specification.
3. A question mark only is displayed when a message is not specified.
4. The [HE] key must be pressed following each data input.
5. Numeric expressions can only be assigned to numeric variables, and string expressions can only be assigned to string variables. A TM error is generated when an attempt is made to assign a string expression to a numeric variable.
6. Quotation marks are not used when entering string data. Enclosing a string in quotation marks causes the quotation marks to be stored as part of the string.
7. Pressing the [CR] key without entering data inputs a string of length 0 for a string variable, while a numeric variable retains its current value.
8. Generally, the logical line immediately following the message is input. The cursor can, however, be moved to any position on the virtual screen (using the cursor keys), and all data from the current cursor position to the end of the current logical line are input when [CR] is pressed.
9. Numeric expressions may be used for numeric value input.
10. Pressing the [IN] key or changing modes during execution of the INPUT statement terminates program execution.
11. Pressing the [IN] key during execution of INPUT leaves program execution.
12. Input data can be edited using the [IN] key, cursor keys, etc.

Character data can be input within the range of character codes 20 through 7E and 80 through FF.

SAMPLE PROGRAM:

10 INPUT ""INPUT STRING"" ; $S
20 PRINT ""$S = "" ; $S
30 END

Displays string entry.
JRPOSE: Assigns a single character input from the keyboard to a variable.

SAMPLE: A$ = INKEYS$

EXPLANATION:
Returns the character or performs the function corresponding to the key pressed during execution of this statement. A null string is returned if a key is not pressed.
The following operations are performed when the keys listed below are pressed during execution of INKEYS$.

ESC: Terminates program execution.

F5: Suspends program execution.

One-key commands and one-key functions: Return a null string.
The cursor is not displayed during data input stand by, and input characters are not displayed. Control codes (00H - 1FH) can be input, but the corresponding operations will not be performed.

EE: INPUT$

AMPLE PROGRAM:

10 PRINT "PRESS ANY KEY" ;
20 C$ = INKEYS$
30 IF C$="" THEN 20
40 CLS: PRINT "YOU PRESS" ; C$ ; "KEY"
50 END

Displays character corresponding to key input.
PURPOSE: Assigns a specified number of characters from the keyboard to a variable.

FORMAT: INPUT$ (number of characters) Numeric expression

EXAMPLE: A$ = INPUTS (3)

PARAMETERS: number of characters: Numeric expression truncated to an integer in the range of 0 ≤ number of characters < 256

EXPLANATION:
1. A string of the length specified by the number of characters is read from the keyboard buffer. Execution waits for the keyboard input when the buffer is empty.
2. The following operations are performed when the keys listed below are pressed during execution of INPUT$.
   - Enter: Halts program execution.
     One-key commands and one-key functions: Return a null string.
3. The cursor is not displayed during data input stand by, and input characters are not displayed. Control codes (&H00 ~ &H1F) can be input, but the corresponding operations will not be performed.

SEE: INKEYS

SAMPLE PROGRAM:

```
10 PRINT "ENTER SECRET CODE" ;
20 IDS = INPUTS (4)
30 IF IDS <> "9876" THEN 10
40 PRINT:PRINT "OK"
```

Checks for validity of input secret code 9876.
**DIM**

**PURPOSE:** Declares an array.

**FORMAT:**

```
DIM array name [ , subscript maximum value [ , subscript maximum value]*]
 Numeric expression
 [ , array name [ , subscript maximum value [ , subscript maximum value]*])*]
 Numeric expression
```

**EXAMPLE:**

```
DIM A$ (10), B$ (10), X (2, 2, 2)
```

**PARAMETERS:**

1. array name: Variable name
2. subscript maximum value: Numeric expression truncated to an integer

**EXPLANATION:**

Declares an array of the dimensions determined by the number of subscript maximum values. The size of the array is determined by each subscript maximum value.

Array elements range from 0 through the specified subscript maximum value.

All elements of a newly declared array are set to their initial value. For numeric arrays, the initial value is 0, while string arrays assigned null strings (length 0).

The size of an array is limited by available memory capacity. Declaration by the DIM statement is subjected to the limitations specified for logical lines (255 characters).

Declaring identical (same array name, same subscript maximum value) in the same program causes second declaration to be disregarded. Declaring two arrays with identical names and different subscript maximum values results in a DD error.

An array variable cannot be used unless they are first declared in a DIM statement.

```plaintext
E: ERASE, CLEAR
```

**SAMPLE PROGRAM:**

```
10 DIM A$ (5)
20 FOR I = 65 TO 70
30 AS (I-65) = CHR$ (I)
40 PRINT A$ (I-65) ;
50 NEXT I
```

Respectively assigns A through F to array cells A$ (0) through A$ (5).
PURPOSE: Erases a specified array.

FORMAT: ERASE [array name [, array name]*]

EXAMPLE: ERASE A$, X

PARAMETERS: array name: Variable name

EXPLANATION:
1. Erases the specified array from memory.
2. An error does not result when the specified array does not exist, and the program proceeds to the next executable statement.
3. The ERASE statement cannot be used in a FOR~NEXT loop.
4. To declare an array using a name already assigned to an existing array, first erase the existing array with the ERASE statement.

SEE: DIM

SAMPLE PROGRAM:

10 CLEAR
20 DIM A$ (10), B$ (10)
30 ERASE A$
40 VARLIST

Declares arrays A$ and B$, and then erases array A$. 
**PEEK**

**Purpose:** Returns the value stored at the specified memory address.

**Syntax:**
```
PEEK ( address )
```

**Numeric expression**

**Example:**
```
PEEK (&H100)
```

**Parameters:**
- `address`: Numeric expression truncated to an integer in the range of 
  $-32769 < \text{address} < 65536$. Negative addresses are added to 65536 and the contents of the resulting address are returned (i.e. `PEEK (-1)` is identical to `PEEK (65535)`).

**Explanation:**
Returns the value stored in memory at the specified address.
The actual address is specified using the DEFSEG statement.

```
DEFSEG = 2
A = PEEK (&H100)
```
The above does not directly read the contents of address `&H100` (256). Instead, the contents of `&H120` (288) are read ($16 \times 2 + 256 = 288$).
Further information of segments can be found under DEFSEG.

**Example Program:**
```
5 DEFSEG = &H0
10 FOR I = &H0C00 TO &H0D00
20 PRINT HEX$(PEEK (I )); " " ;
30 NEXT I
```

Prints memory contents from `&H0C00` to `&H0D00` in hexadecimal.
POKE

PURPOSE:  Writes data to a specified address.

FORMAT:   POKE address, data
           Numeric expression Numeric expression

EXAMPLE:  POKE &H7000, 0

PARAMETERS:  
1. address: Numeric expression truncated to an integer in the range of
   \[-32769 < \text{address} < 65536\]. Negative addresses are added to 65536
   and data are written to the resulting address (i.e. POKE -1, is identical
to POKE 65535, data).
2. data: Numeric expression truncated to an integer in the range of
   \[0 \leq \text{data} < 256\]

EXPLANATION:
1. Writes data to the specified address in memory.
2. Runaway execution may result if the contents of an address outside the user work area
   is altered using the POKE statement.
3. The actual address is specified using the DEFSEG statement.
   DEFSEG = 2
   POKE &H100, 0
   The above does not directly write data to address &H100 (256). Instead, 0 is written to
   address &H120 (288).
   \[16 \times 2 + 256 = 288\]
   Further information of segments can be found under DEFSEG.

SEE:       PEEK, DEFSEG

SAMPLE PROGRAM:

10       DEFSEG = &H0
20       FOR I = &H7000 TO &H7010
30       POKE I, 0
40       NEXT I
50       END

Clears (assigns zeros) memory from 7000H to 7010H.
DEFSEG

PURPOSE: Specifies segment base address.

FORMAT: DEFSEG segment address
        Numeric expression

EXAMPLE: DEFSEG = 16

PARAMETERS: segment address: Integer within the range of -32768 ≤ segment address < 65536

EXPLANATION:
- Specifies the segment base address for use with the PEEK and POKE commands. The
  relationship between the address (offset address) and the segment address within the PEEK
  and POKE format is as follows:
  actual address = segment address × 16 + offset address
- The initial specification for DEFSEG is 0 whenever power is switched ON, or the P button
  or ALL RESET button is pressed.

SEE: PEEK, POKE

AMPLE PROGRAM:

10 DEFSEG = &H1000
20 A = PEEK (&H00F0)

In this case, the value assigned to A is that contained in address 100F0H.
ON ERROR GOTO

PURPOSE: Specifies the line number to which execution branches when an error is generated.

FORMAT: ON ERROR GOTO branch destination line number
Line number

EXAMPLE: ON ERROR GOTO 1000

PARAMETERS: branch destination line number:
Integer in the range of 0 ≤ line number ≤ 65535

EXPLANATION:
1. Specifies the line number to which program execution branches when an error is generated. The program returns to normal operations when a RESUME statement is executed after the error handling routine (starting at the specified line number) is executed.
2. An error is generated and program execution is halted when the branch destination line number is 0.
3. An error generated after execution branches to the specified line number causes an error message to be displayed and program execution to be halted.
4. An ON ERROR GOTO statement must be followed by a corresponding RESUME statement in the same program area. Branching to another program area using ON ERROR GOTO generates an error when the RESUME statement in the other program area is executed.

* The operations outlined are limited to BASIC program execution.

SEE: ERR, ERL, RESUME

SAMPLE PROGRAM:

10 ON ERROR GOTO 40
20 **ERROR**
30 END
40 PRINT "'OOPS! ERROR!!!'" : BEEP 1
50 RESUME 30

Execution of line 40, followed by line 30 if error generated. The program shown here is only an error subroutine and does nothing by itself.
RESUME

PURPOSE: Returns from an error handling routine to the main routine.

FORMAT:

\[
\text{RESUME} \begin{cases} \text{NEXT} \\
\text{return line number} \\
\text{Line number} \end{cases}
\]

AMPLE: RESUME NEXT
        RESUME 100

PARAMETERS: 1. NEXT
             2. return line number: Integer in the range of \(1 \leq \text{line number} \leq 65535\)

PLANATION:
This statement is entered at the end of an error handling routine.
The statement that generated the original error is the default option when the return destination (NEXT or return line number) is omitted.
Program execution returns to the statement following the statement that generated the original error when NEXT is specified.
Return line number specifies the line to which program execution is to be resumed.
A RESUME statement without a return destination or a RESUME statement that specifies the line in which the original error was generated as the return line number cannot be written at the beginning of the error handling routine. This would result in an endless loop between the statement in which the error was generated and the error handling routine.
A RESUME statement must always be included in the same program area as the ON ERROR GOTO statement.

E: ERR, ERL, ON ERROR GOTO

AMPLE PROGRAM:

10 ON ERROR GOTO 1000
20 INPUT A
30 D = 1/A
40 PRINT "1/" ; A ; "=" ; D
50 GOTO 20
1000 PRINT "0 IS ILLEGAL"
1010 RESUME 20

Calculates reciprocals of input values and returns to line 20 if a 0 is entered (resulting in division by 0).
**ERL**

**PURPOSE:** Returns the number of a line in which an error has been generated.

**FORMAT:** \( \text{ER} = \text{ERL} \)

**EXPLANATION:**
The value of ERL can only be changed within a program, and the value is cleared when a program is executed or when the power of the unit is switched OFF.

**SEE:** ERR, ON ERROR GOTO

**SAMPLE PROGRAM:**

```
10 ON ERROR GOTO 40
20 ***ERROR***
30 END
40 PRINT "ERROR LINE=" ; ERL
50 RESUME 30
```

Error is generated in line 20 and corresponding error code is displayed in line 40.

---

**ERR**

**PURPOSE:** Returns the error code which corresponds to a generated error.

**FORMAT:** \( \text{PRINT ERR} \)

**EXPLANATION:**
The value of ERR can only be changed within a program, and the value is cleared when a program is executed or when the power of the unit is switched ON. See the error message table on page 397 for details concerning error codes and their corresponding error messages.

**SEE:** ON ERROR GOTO, ERL, Error Message Table

**SAMPLE PROGRAM:**

```
10 ON ERROR GOTO 40
20 ***ERROR***
30 END
40 PRINT "ERROR CODE=" ; ERR
50 RESUME 30
```

An error is generated in line 20 and the corresponding error code is displayed in line 40.
ANGLE

JRPOSE: Specifies the angle unit.

ORMAT: ANGLE angle specification
        Numeric expression

XAMPLE: ANGLE 0

ARAMETERS: angle specification: Numeric expression truncated to an integer in the range of 0 ≤ angle specification < 3

XPLANATION:
The angle units for the trigonometric function can be specified using the values 0, 1, and 2.

0: DEG (degrees)
1: RAD (radians)
2: GRAD (grads)

The relationships between the angle units are as follows:

<table>
<thead>
<tr>
<th>Angle Unit</th>
<th>DEG</th>
<th>RAD</th>
<th>GRAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1DEG =</td>
<td>1</td>
<td>$\frac{\pi}{180}$</td>
<td>100</td>
</tr>
<tr>
<td>1RAD =</td>
<td>$\frac{180}{\pi}$</td>
<td>1</td>
<td>200</td>
</tr>
<tr>
<td>1GRAD =</td>
<td>$\frac{90}{100}$</td>
<td>$\frac{\pi}{200}$</td>
<td>1</td>
</tr>
</tbody>
</table>

$90^\circ = \frac{\pi}{2}$ rad = 100 grad

3. ANGLE 0 is set whenever NEW ALL is executed.
4. The angle unit can also be specified using the $\equiv$ key.

SAMPLE PROGRAMS:

```
10 ANGLE 0 'DEGREE
20 PRINT SIN 30 ;
30 ANGLE 1 'RADIAN
40 PRINT SIN (PI/6) ;
50 ANGLE 2 'GRAD
60 PRINT SIN (100/3)
```

Calculates and displays sin 30° in the degree mode, sin $\frac{\pi}{6}$ in the radian mode, and sin $\frac{100}{3}$ in the grad mode.
SIN COS TAN

PURPOSE: Returns the value of the corresponding trigonometric function value for the argument.

FORMAT:  

SIN (argument)  
   Numeric expression

COS (argument)  
   Numeric expression

TAN (argument)  
   Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

EXAMPLE: SIN (30), COS (PI/2)

PARAMETERS: argument: Numeric expression (angle)
   | argument | < 1440 (DEG)
   | argument | < 8π (RAD)
   | argument | < 1600 (GRAD)

EXPLANATION: Returns the value of the corresponding trigonometric function for the argument.

SIN SINE
COS COSINE
TAN TANGENT

SEE: ANGLE, ASN, ACS, ATN

SAMPLE PROGRAM:

10 ANGLE 0
20 INPUT "DEGREE = " , D
30 PRINT "SIN (" ; D ; " )=" ; SIN D
40 PRINT "COS (" ; D ; " )=" ; COS D
50 PRINT "TAN (" ; D ; " )=" ; TAN D
60 GOTO 20

Displays trigonometric function values for input angles.
ASN
ACS
ATN

PURPOSE: Returns the value of the corresponding inverse trigonometric function for the argument.

FORMAT: 

\[
\begin{align*}
\text{ASN} & \quad (\text{argument}) \\
& \quad \text{Numeric expression} \\
\text{ACS} & \quad (\text{argument}) \\
& \quad \text{Numeric expression} \\
\text{ATN} & \quad (\text{argument}) \\
& \quad \text{Numeric expression}
\end{align*}
\]

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

SAMPLE: 

ASN (0.1)

PARAMETERS: argument: Numeric expression in the range of \(-1 \leq \text{argument} \leq 1\) (ASN, ACS)

EXPLANATION: 

Returns the value of the corresponding inverse trigonometric function for the argument.

ASN ARCSINE
ACS ARCCOSINE
ATN ARCTANGENT

Function values are returned within the following ranges:

- \(-90^\circ \leq \text{ASN} (x) \leq 90^\circ, 0^\circ \leq \text{ACS} (x) \leq 180^\circ\)
- \(-90^\circ \leq \text{ATN} (x) \leq 90^\circ\)

SEE: ANGLE, SIN, COS, TAN

SAMPLE PROGRAM:

10 ANGLE 1
20 INPUT "INPUT NUMBER (-1 TO 1)" ; N
30 PRINT N ; " = S I N (" ; ASN N ; "° R A D)"
40 PRINT N ; " = C O S (" ; ACS N ; "° R A D)"
50 PRINT N ; " = T A N (" ; ATN N ; "° R A D)"
60 ANGLE 0 : END

Displays trigonometric angles in radians for each input in range of \(-1\) to 1.
HYP SIN
HYP COS
HYP TAN

PURPOSE: Returns the value of the corresponding hyperbolic function for the argument.

FORMAT: 

HYP SIN (argument)
Numeric expression

HYP COS (argument)
Numeric expression

HYP TAN (argument)
Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

EXAMPLE: HYP SIN (1.5)

PARAMETERS: argument: Numeric expression
HYP SIN |argument| ≤ 230.2585092
HYP COS |argument| ≤ 230.2585092

EXPLANATION:
Returns the value of the corresponding hyperbolic function for the argument.

HYP SIN (x) : sinh x = (e^x - e^-x)/2
HYP COS (x) : cosh x = (e^x + e^-x)/2
HYP TAN (x) : tanh x = (e^x - e^-x)/(e^x + e^-x)

SEE: HYP ASN, HYP ACS, HYP ATN

SAMPLE PROGRAM:

10 INPUT "INPUT NUMBER (UP TO 230)" ; N
20 PRINT "HSN (" ; N ; ")=" ; HYSIN N
30 PRINT "HCS (" ; N ; ")=" ; HYP COS N
40 PRINT "HTN (" ; N ; ")=" ; HYP TAN N
50 END

Displays the hyperbolic functions for numeric input up to 230.
HYP ASN
HYP ACS
HYP ATN

PURPOSE: Returns the value of the corresponding inverse hyperbolic function for the argument.

FORMAT:

HYP ASN (argument)
                          Numeric expression

HYP ACS (argument)
                          Numeric expression

HYP ATN (argument)
                          Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

SAMPLE:

HYP ASN (10)

PARAMETERS: argument: Numeric expression

HYP ASN |argument| < 5 × 10^99 (5E+99)
HYP ACS  1 ≤ argument < 5 × 10^99 (5E+99)
HYP ATN  -1 < argument < 1

EXPLANATION:

Returns the value of the corresponding inverse hyperbolic function for the argument.

HYP ASN (x) : sinh^-1 x = loge (x + sqrt(x^2 + 1))

HYP ACS (x) : cosh^-1 x = loge (x + sqrt(x^2 - 1))

HYP ATN (x) : tanh^-1 x = 1/2 loge (1 + x) / (1 - x)

SEE:

HYP SIN, HYP COS, HYP TAN

SAMPLE PROGRAM:

10  INPUT "INPUT NUMBER (1 OR GREATER)" ; N
20  PRINT "HAS (" ; N ; ")=" ; HYPASN N
30  PRINT "HAC (" ; N ; ")=" ; HYPACS N
40  END

Displays inverse hyperbolic function value for numeric input of 1 or greater.
EXP

PURPOSE: Returns the value of the exponential function for the argument.

FORMAT: EXP (argument)
         Numeric expression

         * The parentheses enclosing the argument can be omitted when the
           argument is a numeric value or variable.

EXAMPLE: EXP (1)

PARAMETERS: argument: Numeric expression in the range of argument ≤ 230.2585092

EXPLANATION:
Returns the value of the exponential function value for the argument.
EXP (x) = e^x

SEE: LOG, LN

SAMPLE PROGRAM:

10 INPUT "e^X (UP TO 230)" ; N
20 PRINT "e^" ; N ; " = " ; EXP N
30 END

Displays exponential function value for numeric input up to 230.
LOG
LN

PURPOSE:  Returns the value of the corresponding logarithm function for the argument.

FORMAT:   LOG   (argument)
          Numeric expression
LN   (argument)
          Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

SAMPLE:   LOG (2), LN (3)

PARAMETERS:  argument:  Numeric expression
              LOG :  0 < argument
              LN :  0 < argument

EXPLANATION:
Returns the value of the corresponding logarithm function value for the argument.

LOG :  Common logarithm  log_{10} x, log x
LN  :  Natural logarithm  log e x, ln x

SAMPLE PROGRAM:

10 INPUT "INPUT NUMBER" ; N
20 PRINT "LOG" ; N ; " = " ; LOG N
30 PRINT "LN" ; N ; " = " ; LN N
40 END

Displays logarithm function values for numeric input greater than 0.
### SQR

**PURPOSE:**
Returns the square root of the argument.

**FORMAT:**
```plaintext
SQR (argument)
```
- Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

**EXAMPLE:**
```plaintext
SQR (4)
```

**PARAMETERS:**
- argument: Numeric expression in the range of $0 \leq \text{argument}$

**EXPLANATION:**
Returns the square root of the argument.

```plaintext
SQR (x) : \sqrt{x}
```

**SAMPLE PROGRAM:**
```plaintext
10 FOR I = 0 TO 10
20 PRINT "SQR" ; I ; SQR I
30 NEXT I
40 END
```

Displays square roots of values from 0 through 10.

### CUR

**PURPOSE:**
Returns cube root of argument.

**FORMAT:**
```plaintext
CUR (argument)
```
- Numeric expression

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

**EXAMPLE:**
```plaintext
X = CUR (Y)
```

**PARAMETERS:**
- argument: Numeric expression

**EXPLANATION:**
Returns the cube root of the argument.

```plaintext
CUR (x) : \sqrt[3]{x}
```

**SAMPLE PROGRAM:**
```plaintext
10 A = 27
20 PRINT A ; ", CUBE ROOT" ; CUR A
30 END
```

Returns cube root of value assigned to variable A.
PURPOSE: Returns the absolute value of the argument.

FORMAT: \[
\text{ABS (argument)} \\
\text{Numeric expression}
\]
* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

SAMPLE: \[
\text{ABS (-1.5)}
\]

PARAMETERS: argument: Numeric expression

EXPLANATION:
\text{Returns the absolute value of the argument.}

\text{ABS (x) : } |x|$

SAMPLE PROGRAM:

10 INPUT "INPUT NUMBERS" ; N
20 A = ABS N
30 PRINT N ; "ABS ( )=" ; A
40 END

Displays the absolute value of an input value.
SGN

PURPOSE: Returns a value which corresponds to the sign of the argument.

FORMAT: \[ \text{SGN} \ (\text{argument}) \]

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

EXAMPLE: \text{SGN (A)}

PARAMETERS: argument: Numeric expression

EXPLANATION: Returns a value of \(-1\) when the argument is negative, 0 when the argument equals 0, and 1 when the argument is positive.

<table>
<thead>
<tr>
<th>Argument (X)</th>
<th>SGN (X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(X &lt; 0)</td>
<td>(-1)</td>
</tr>
<tr>
<td>(X = 0)</td>
<td>0</td>
</tr>
<tr>
<td>(X &gt; 0)</td>
<td>1</td>
</tr>
</tbody>
</table>

SAMPLE PROGRAM:

10 INPUT "INPUT NUMBER" ; N
20 \(S = \text{SGN} \ N\)
30 IF \(S < 0\) THEN PRINT "NOT ZERO" : END
40 PRINT "ZERO" : END

Uses SGN function to determine whether or not an input value equals 0.
**INT**

**PURPOSE:** Returns the largest integer which does not exceed the value of the argument.

**FORMAT:**

\[
\text{INT } \frac{\text{argument}}{\text{Numeric expression}}
\]

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

**SAMPLE:**

\[
\text{INT (1.3)}
\]

**PARAMETERS:** argument: Numeric expression

**EXPLANATION:**
Returns the largest integer which does not exceed the value of the argument.

**SEE:**

FIX, FRAC

**SAMPLE PROGRAM:**

\[
\begin{align*}
10 & \text{ FOR } I = 1 \text{ TO } 10 \\
20 & N = \text{RAN} * 10 \\
30 & \text{LPRINT } "\text{INT (}" ; N ; "\text{)}=" ; \text{INT } N \\
40 & \text{NEXT } I \\
50 & \text{END}
\end{align*}
\]

Converts random values to integers and outputs results to printer.

**FIX**

**PURPOSE:** Returns the integer part of the argument.

**FORMAT:**

\[
\text{FIX } \frac{\text{argument}}{\text{Numeric expression}}
\]

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

**SAMPLE:**

\[
\text{FIX (-1.5)}
\]

**PARAMETERS:** argument: Numeric expression

**EXPLANATION:**
Returns the integer part of the argument.

**SEE:**

INT

**SAMPLE PROGRAM:**

\[
\begin{align*}
10 & \text{ INPUT } A \\
20 & \text{PRINT } "\text{FIX (}" ; A ; "\text{)}=" ; \text{FIX } A \\
30 & \text{GOTO } 10
\end{align*}
\]

Displays the integer part of input values.
FRAC

PURPOSE: Returns the fractional part of the argument.

FORMAT: \[
\text{FRAC (argument)}
\]
\[
\text{Numeric expression}
\]
* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

EXAMPLE: \[
\text{FRAC (3.14)}
\]

PARAMETERS: argument: Numeric expression

EXPLANATION:
1. Returns the fractional part of the argument.
2. The sign (±) of the value is the same as that for the argument.

SAMPLE PROGRAM:

\[
10 \text{ FOR I = 1 TO 10}
20 \text{ N = RAN# * 10}
30 \text{ LPRINT ""FRAC ("" ; N ; ")"" = "" ; FRAC N}
40 \text{ NEXT I}
50 \text{ END}
\]

Isolates fractional parts of random values and outputs results to printer.

ROUND

PURPOSE: Rounds the argument at the specified digit.

FORMAT: \[
\text{ROUND (argument, digit)}
\]

EXAMPLE: \[
\text{ROUND (A, -3)}
\]

PARAMETERS: 1. argument: Numeric expression
2. digit: Numeric expression truncated to an integer in the range of \(-100 < \text{digit} < 100\)

EXPLANATION:
Rounds the argument (to the nearest whole number) at the specified digit.

SAMPLE PROGRAM:

\[
10 \text{ N = RAN# * 1000}
20 \text{ PRINT N}
30 \text{ INPUT ""WHERE"" ; R}
40 \text{ PRINT ROUND (N, R)}
50 \text{ END}
\]

Displays random value and then rounds value at digit specified by numeric input.
For example, responding to prompt "WHERE" with input of \(-2\) when \(N = 610.5765383\) produces result of \(610.6\).
**PURPOSE:** Returns a random value in the range of 0 to 1.

**FORMAT:**

\[
\text{RAN}\# \quad \text{(argument)}
\]

\[
\text{Numeric expression}
\]

* The parentheses enclosing the argument can be omitted when the argument is a numeric value or variable.

**EXAMPLE:**

\[
\text{RAN}\# * 10
\]

**PARAMETERS:** argument: Numeric expression

**EXPLANATION:**

Returns a random value in the range of 0 to 1. \((0 < \text{RAN}\#(X) < 1)\)

Random numbers are generated from the same table when \(X = 1\).

The last random number generated is repeated when \(X = 0\).

Random numbers are generated from the random table when \(X = -1\).

Random number generation begins with the same value each time a program is executed. This means that the same series of numbers is generated unless the argument of \(\text{RAN}\#\) is omitted or is equal to \(-1\).

**SAMPLE PROGRAM:**

10 \(R = \text{RAN}\# (1)\) : PRINT \(R\)
20 \(R = \text{RAN}\# (0)\) : PRINT \(R\)
30 \(R = \text{RAN}\# (-1)\) : PRINT \(R\)
40 GOTO 10

Generates random numbers using each type (positive, negative, zero) of argument.
**PI**

**PURPOSE:** Returns the value of \( \pi \).

**FORMAT:** \( \text{PI} \)

**EXAMPLE:** \( S = 2 \times \text{PI} \times R \)

**EXPLANATION:**
1. Returns the value of \( \pi \).
2. The value of \( \pi \) used for internal calculations is 3.1415926536.
3. The displayed value is rounded off to 10 digits, so the value of \( \pi \) is displayed as 3.141592654.

**SAMPLE PROGRAM:**

```
10 INPUT "RADIUS" ; R
20 PRINT "CIRCUMFERENCE = " ; 2 * PI * R
30 PRINT "AREA = " ; R ^ 2 * PI
40 END
```

Calculates circumference and area of circle after input of radius.

**FACT**

**PURPOSE:** Returns factorial of argument.

**FORMAT:** \( \text{FACT} \ (\text{argument}) \)

**EXAMPLE:** \( A = \text{FACT} (10) \)

**PARAMETER:** argument : Integer in the range of 0 \( \leq \) argument \( \leq \) 69

**EXPLANATION:**
1. Returns the factorial of the argument.
   \( \text{FACT} x : x! \)
2. A fractional value as the argument generates an error.

**SAMPLE PROGRAM:**

```
10 X = 5
20 Y = FACT X
30 PRINT X ; "! = " ; Y
```

Assigns factorial of the value of variable \( X \) to variable \( Y \) and displays the result.
**NPR**

**PURPOSE:** Returns the permutation nPr for the values of n and r.

**FORMAT:**

\[
\text{NPR} \left( \frac{n \text{ value}}{\text{Numeric expression}}, \frac{r \text{ value}}{\text{Numeric expression}} \right)
\]

**SAMPLE:**

\[ X = \text{NPR} \left( 69, 20 \right) \]

**PARAMETERS:**

\[
\begin{align*}
\text{n : } & \quad \text{Integer in the range of } 0 \leq r \leq n < 10^{10} \\
\text{r : } &
\end{align*}
\]

**EXPLANATION:**

Returns the permutation:

\[
\text{nPr} = \frac{n!}{(n - r)!}
\]

A fractional value as either n or r generates an error.

**SAMPLE PROGRAM:**

```plaintext
10 \text{N} = 10 : \text{P} = 5 \\
20 \text{X} = \text{NPR} \left( \text{N}, \text{R} \right) \\
30 \text{PRINT X}
```

Calculates \(10!P_5\) and displays the result.

---

**NCR**

**PURPOSE:** Returns the combination nCr for the values of n and r.

**FORMAT:**

\[
\text{NCR} \left( \frac{n \text{ value}}{\text{Numeric expression}}, \frac{r \text{ value}}{\text{Numeric expression}} \right)
\]

**SAMPLE:**

\[ X = \text{NCR} \left( 70, 35 \right) \]

**PARAMETERS:**

\[
\begin{align*}
\text{n : } & \quad \text{Integer in the range of } 0 \leq r \leq n < 10^{10} \\
\text{r : } &
\end{align*}
\]

**EXPLANATION:**

Returns the combination:

\[
\text{nCr} = \frac{n!}{r! \left( n - r \right)!}
\]

A fractional value as either n or r generates an error.

**SAMPLE PROGRAM:**

```plaintext
10 \text{N} = 8 : \text{R} = 4 \\
20 \text{X} = \text{NCR} \left( \text{N}, \text{R} \right) \\
30 \text{PRINT X}
```

Calculates \(8C_4\) and displays the result.
PURPOSE: Converts rectangular coordinates \((x, y)\) to polar coordinates \((r, \theta)\).

FORMAT: \[
\text{POL} \quad \left( \frac{x\text{-coordinate}}{\text{Numeric expression}}, \frac{y\text{-coordinate}}{\text{Numeric expression}} \right)
\]

EXAMPLE: \[
\text{POL} (3, 2)
\]

PARAMETERS: x-coordinate : Numeric expression  
y-coordinate : Numeric expression. \(|x| + |y| > 0\)

EXPLANATION:

1. Converts rectangular coordinates \((x, y)\) into polar coordinates \((r, \theta)\). The following relational expressions are used at this time:

\[
\begin{align*}
    r &= \sqrt{x^2 + y^2} \\
    \cos \theta &= \frac{x}{\sqrt{x^2 + y^2}} \\
    \sin \theta &= \frac{y}{\sqrt{x^2 + y^2}}
\end{align*}
\]

2. The value of \(r\) is automatically assigned to variable \(X\), while \(\theta\) is automatically assigned to variable \(Y\).

3. The value of \(\theta\) is given as follows:
   - \(180^\circ < \theta \leq 180^\circ\) (DEG)
   - \(-\pi < \theta \leq \pi\) (RAD)
   - \(-200\)\(^{\text{grads}} < \theta \leq 200\)\(^{\text{grads}}\) (GRA)

SAMPLE PROGRAM:

\[
\begin{align*}
    10 & \quad \text{A} = 5 : \text{B} = 3 \\
    20 & \quad \text{Z} = \text{POL} (\text{A}, \text{B}) \\
    30 & \quad \text{PRINT} \; X ; Y
\end{align*}
\]

Converts rectangular coordinate point \((5, 3)\) to polar coordinates.
PURPOSE: Converts polar coordinates $(r, \theta)$ to rectangular coordinates $(x, y)$.

FORMAT: 

\[
\text{REC} \left( \begin{array}{c} \text{distance } r \text{ (numeric expression)} \\ \text{angle } \theta \text{ (numeric expression)} \end{array} \right)
\]

SAMPLE: 

\[
\text{REC} (10, 15)
\]

PARAMETERS: 

- distance $r$: $0 \leq r < 10^{100}$
- angle $\theta$: $-1440^\circ < \theta < 1440^\circ$ (DEG)
  $-8\pi < \theta < 8\pi$ (RAD)
  $-1600$ (grads) $< \theta < 1600$ (grads) (GRA)

EXPLANATION: Converts polar coordinates $(r, \theta)$ to rectangular coordinates $(x, y)$. The following relational expressions are used at this time:

- $x = r \cos \theta$, $y = r \sin \theta$

The value of $x$ is automatically assigned to variable $X$, while $y$ is automatically assigned to variable $Y$.

SAMPLE PROGRAM:

10 A = 2 : B = 30
20 Z = REC (A, B)
30 PRINT X ; Y

Converts polar coordinate point $(2, 30)$ to rectangular coordinates.
CHR$:

PURPOSE: Returns a single character which corresponds to the specified character code.

FORMAT: CHRS $( code )

EXAMPLE: CHRS $ (65)

PARAMETERS: code: Numeric expression truncated to an integer in the range of $0 \leq code < 256$

EXPLANATION: Variables can also be used as a parameter, and decimal parts of numeric values are truncated. A null is returned when a character does not exist for the specified character code.

SEE: ASC

SAMPLE PROGRAM:

```
10 FOR I = 65 TO 90
20 PRINT CHRS $(I);
30 NEXT I
```

Displays characters from 65 through 90 in character code.
**PURPOSE:** Returns the character code corresponding to the character in the first (leftmost) position of a string.

**FORMAT:**

\[
\text{ASC } (\text{ string }) \]

String expression

**EXAMPLE:**

\[
\text{ASC } ("A")
\]

**PARAMETERS:**

| string: String expression |

**EXPLANATION:**

Returns the character code corresponding to a character. The character code for the first (leftmost) character only is returned for a string of two or more characters long.

A value of 0 is returned for a null string.

**SEE:** CHRS, Character Code Table

**SAMPLE PROGRAM:**

10 INPUT "INPUT CHARACTERS" ; A$
20 BS = LEFTS (A$,1)
30 C = ASC (A$)
40 PRINT "FIRST CHAR=" ; BS ; "CODE=" ; C
50 END

Displays first character and corresponding character code for string input.
PURPOSE: Converts the argument (numeric value or numeric expression value) to a string.

FORMAT: STR$ (argument)
        String expression

EXAMPLE: STR$ (123), STR$ (255 + 3)

PARAMETERS: argument: Numeric expression

EXPLANATION:
1. Converts decimal values specified in the argument to strings.
2. Converted positive values include a leading space and converted negative values are preceded by a minus sign.

SEE: VAL

SAMPLE PROGRAM:

10 INPUT "INPUT NUMBERS" ; N
20 SS = STRS (N)
30 CS = MID$ (SS, 2, 1)
40 PRINT "FIRST CHARACTER = " ; CS
50 END

Converts numeric input to a string. Next, the first number of converted string is displayed as character.
**VAL**

**PURPOSE:** Converts a numeric character string to a numeric value.

**FORMAT:**

\[
\text{VAL (string)}
\]

String expression

**SAMPLE:**

\[A = \text{VAL ("345")}\]

**PARAMETERS:**

- **string:** String expression

**EXPLANATION:**

Converts a numeric character string to a numeric value.

Numeric characters are converted up to the point in the string that a non-numeric character is encountered. All subsequent characters are disregarded from the non-numeric character onwards. (i.e. when \(A = \text{VAL ("123A456")}, A = 123\). The value of this function becomes 0 when the length of the string is 0 or when the leading character is non-numeric.

**SEE:**

**SAMPLE PROGRAM:**

```
10 INPUT "VALUE1", A$
20 INPUT "VALUE2", B$
30 CS = A$ + B$
40 C = \text{VAL (A$)} + \text{VAL (B$)}
50 PRINT CS, C
```

Performs string addition and numeric addition of two input strings.
VALF

PURPOSE: Performs calculation of numeric expression expressed as string, and returns the result.

FORMAT: VALF ( string )
         String expression

EXAMPLE: VALF (XS)

PARAMETERS: string: String expression

EXPLANATION:
1. Performs calculation of numeric expressions which are expressed as strings, and returns their results.
2. An error is generated when an intermediate or final result of calculation exceeds $10^{100}$.
3. VALF cannot be used within a VALF argument.

SAMPLE PROGRAM:

10 XS = "'123 + 456'"
20 PRINT VALF (XS)
30 PRINT VALF ("'EXP 2'")

RUN
579
7.389056099

Executes strings "'123 + 456'" and "'EXP 2'" as numeric expressions and displays results.
**MID$**

**PURPOSE:** Returns a substring of a specified length from a specified position within a string.

**FORMAT:**

\[ \text{MID$ ( string , position , number of characters )} \]

String expression   Numeric expression   Numeric expression

**SAMPLE:**

MID$ (A$, 5, 3)

**PARAMETERS:**

1. **string:** String expression
2. **position:** Numeric expression truncated to an integer in the range of 1 ≤ position < 256
3. **number of characters:** Numeric expression truncated to an integer in the range of 0 ≤ number of characters < 256. The default option is from the specified position to the end of the string when this parameter is omitted.

**EXPLANATION:**

Returns a substring of a specified length from a specified position within a string. A substring from the specified position to the end of the string is returned when the length of the substring is not specified.

A substring of length 0 (null) is returned when the specified position exceeds the length of the string.

A substring from the specified position to the end of the string is returned when the specified number of characters is greater than the number of characters from the specified position to the end of the string.

**SEE:** RIGHTS, LEFT$

**SAMPLE PROGRAM:**

```
10 A$ = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
20 INPUT "1 TO 26 FROM" ; B
30 PRINT "1 TO" ; 27 - B ; "TO" ;
40 INPUT E
50 SS = MIDS (A$, B, E)
60 PRINT SS
70 END
```

Uses numeric input to produce alphabetic series of a specified number of characters starting from a specified location.
RIGHTS$  

PURPOSE: Returns a substring of a specified length counting from the right of a string.

FORMAT:  

RIGHTS$ ( string ,      number of characters )      
               String expression          Numeric expression

EXAMPLE: RIGHTS$ ("ABCDEF", 3)

PARAMETERS:  

1. string: String expression
2. number of characters: Numeric expression truncated to an integer in the range of 0 ≤ number of characters < 256.

EXPLANATION:
1. Returns a substring of a specified length counting from the right of string.
2. The entire string is returned as the substring when the specified number of characters is greater than the number of characters in the string.

SEE: MIDS$, LEFT$

SAMPLE PROGRAM:

10  A$ = "ABCDEFGHIJKLMNOPQRSTUVWXYZ" 
20  PRINT A$ 
30  INPUT "1 TO 26 HOW MANY GET" ; N 
40  PRINT RIGHTS$ (A$, N) 
50  END

Uses numeric input to display specified number of characters from end of alphabetic sequence.
PUPPOSE: Returns a substring of a specified length counting from the left of a string.

FORMAT: LEFT$ ( string , number of characters )
         String expression   Numeric expression

SAMPLE: LEFT$ ("ABCDEF", 3)

PARAMETERS: 1. string: String expression
              2. number of characters: Numeric expression truncated to an integer in
                              the range of 0 ≤ number of characters < 256.

EXPLANATION:
Returns a substring of a specified length counting from the left of string.
The entire string is returned as the substring when the specified number of characters
is greater than the number of characters in the string.

SEE: MIDS, RIGHT$

SAMPLE PROGRAM:

10 AS = "ABCDFGHJKLMNPQRSTUVWXYZ"
20 PRINT AS
30 INPUT "1 TO 26 HOW MANY GET" ; N
40 PRINT LEFT$ (AS, N)
50 END

Uses numeric input to display specified number of characters from
beginning of alphabetic sequence.
LEN

PURPOSE: Returns a value which represents the number of characters contained in a string.

FORMAT: \[\text{LEN } (\text{string})\]

EXAMPLE: \(\text{LEN (A$)}\)

PARAMETERS: string: String expression

EXPLANATION: Returns a value which represents the number of character contained in a string, including characters that don't appear on the display (character codes from \&H0 - \&H1F) and spaces.

SAMPLE PROGRAM:

10 INPUT "INPUT CHARACTERS" ; CS
20 PRINT "LENGTH = " ; LEN (CS)
30 END

Determines the length of an input string.

HEX$  

PURPOSE: Returns a hexadecimal string for a decimal value specified in the argument.

FORMAT: \[\text{HEX$ } (\text{argument})\]

EXAMPLE: \(\text{HEX$ (15)}\)

PARAMETERS: argument: Numeric expression truncated to an integer in the range of \(-32769 < \text{argument} < 65536\). Values more than 32767 are converted by subtracting 65536.

EXPLANATION: Returns a 4-digit hexadecimal string for a decimal value specified in the argument.

SEE: &H

SAMPLE PROGRAM:

10 PRINT "DECIMAL" ; TAB (10) ; "HEX"
20 FOR I = 0 TO 16
30 PRINT I ;
40 PRINT TAB (10) ; HEX$ (I) ; : PRINT
50 FOR J = 0 TO 250 : NEXT J
60 NEXT I
70 END

Displays the decimal values from 0 through 16 along with their hexadecimal equivalents.
&H

PURPOSE: Converts the 1 through 4-digit hexadecimal value following &H to a decimal value.

FORMAT: 

<table>
<thead>
<tr>
<th>&amp;H</th>
<th>argument</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hexadecimal value</td>
</tr>
</tbody>
</table>

SAMPLE: A = &HAF

PARAMETERS: 0 ≤ argument ≤ FFFFw

EXPLANATION:
The hexadecimal value is expressed using values 0 through 9, plus characters A through F. In the manual mode, &H is entered followed by the hexadecimal value. Pressing &H produces the decimal equivalent.

Example: &H1B7F &H → 7039

The following shows a typical application within a program. Since a numeric variable cannot be used following &H, the hexadecimal value is appended to &H as a string and then converted to a decimal value using the VAL function.

SEE: HEXS

SAMPLE PROGRAM:

```plaintext
10 REM &H SAMPLE
20 INPUT "&H" ; AS
30 H = VAL("&H" + AS)
40 PRINT "&H" ; AS ; "=" ; H
50 GOTO 10
```

Converts an entered hexadecimal value (4 digits max) to a decimal value. Program execution is terminated using the &H key.
DEG

PURPOSE: Converts a sexagesimal value to a decimal value.

FORMAT: \[
\text{DEG} \left( \frac{\text{degrees}}{\text{Numeric expression}}, \frac{\text{minutes}}{\text{Numeric expression}}, \frac{\text{seconds}}{\text{Numeric expression}} \right)
\]

EXAMPLE: \[
\text{DEG} \left( \frac{1}{\text{Numeric expression}}, \frac{30}{\text{Numeric expression}}, \frac{10}{\text{Numeric expression}} \right)
\]

PARAMETERS: Degrees, minutes, seconds: |DEG (degrees, minutes, seconds)| < 10^100

EXPLANATION:
Converts the degrees, minutes, and seconds of sexagesimal values to decimal values as follow:
\[
\text{DEG} \left( \text{degrees}, \text{minutes}, \text{seconds} \right) = \text{degrees} + \frac{\text{minutes}}{60} + \frac{\text{seconds}}{3600}
\]

SAMPLE PROGRAM:

```plaintext
10 INPUT "DEGREES=", A
20 INPUT "MINUTES=", B
30 INPUT "SECONDS=", C
40 D = DEG (A, B, C)
50 PRINT D
60 END
```

Converts values entered for degrees, minutes, and seconds into a decimal value.
**PURPOSE:** Converts a decimal value to a sexagesimal string.

**FORMAT:**

\[
\text{DMSS} \; (\text{argument})
\]

- Numeric expression

**EXAMPLE:**

DMSS (1.52)

**PARAMETERS:**

- argument:
  - Numeric expression in the range of \(|\text{numeric expression}| < 10^{100}\)

**EXPLANATION:**

Converts decimal values to sexagesimal strings.

Minutes and seconds are not displayed when the argument is in the range of numeric expression \(\geq 1 \times 10^6 (1\text{E6})\). In this case, the absolute value of the input value is converted to a string as it is.

**SAMPLE PROGRAM:**

```
10 INPUT "INPUT NUMBER" ; N
20 PRINT "= " ; DMSS (N)
30 END
```

Converts input decimal values to sexagesimal strings.
I/O COMMANDS

LLIST

PURPOSE: Outputs program contents to the printer.

FORMAT: LLIST \{ [starting line number] [ - ] [ending line number] \}

\{ Line number \}

\{ [ALL] \}

EXAMPLE: LLIST 50 - 100

PARAMETERS: Both the starting line number and ending line number are within the range of 1 \leq\ line number \leq 65535. The last line number used by BASIC is specified when "ALL" is used.

1. starting line number: Program line number from which program content printout is to begin. The default option is the first line of the program.
2. ending line number: Program line number at which program content printout is to end. The default option is the last line of the program.
3. Specifying ALL sequentially outputs all program contents in areas P0 through P9.

EXPLANATION:
1. Outputs program contents to the printer within the specified range.
2. This statement differs from LIST in that output is to the printer without showing program contents on the display.
3. LLIST cannot be used in the CAL mode.

SAMPLE EXECUTION: LLIST ALL

Outputs contents of current program area to printer.
LPRINT

PURPOSE: Outputs text to the printer.

FORMAT: LPRINT [output data] [TAB (numeric expression)]

Output data: Numeric expression

SAMPLE: LPRINT A, B

PARAMETERS: output data: Output control function, numeric expression, or string expression

EXPLANATION:

Outputs data to the printer. When the output data is a control function, the corresponding operation is performed. Numeric or string expressions as output data result in printout of the resulting value.

Numeric expression values are printed in decimal, and the print format is the same as that for the PRINT statement (see PRINT).

String expression values are output as they are to the printer.

Inserting a comma between output data causes a zone tab to be inserted between output data at output.

The tabs are set at 14-character intervals (counting from 0, within a range of 255 characters) following the last carrier return instruction, and zone tab outputs spaces from the current location to the next zone tab. Consequently, the printing of the first character of output data following a comma is performed at the next zone tab.

10 LPRINT
20 FOR I = 1 TO 20 : LPRINT "*", : NEXT I
30 LPRINT
40 END

Inserting a semicolon between output data causes the output data to be output sequentially.

10 LPRINT
20 FOR I = 1 TO 50
30 LPRINT "'" ; I ; "'
40 NEXT I
50 LPRINT
60 END

Inserting a semicolon at the end of the statement causes the location immediately following printout of the last output data to be the next printing position.

Inserting a comma at the end of an LPRINT statement performs a zone tab following printout of the last output data.

Carrier return is performed when a semicolon or comma is not included at the end of the statement. Print positions are counted from 0 through 255, and the count is reset to 0 when it exceeds 255. Zone tabs and the TAB function are performed in accordance with the print position count. CR-LF (internal code 0Dh, 0Ah) is performed at this time.
9. Actual printing begins when a carrier return/line feed code is sent, and carrier return/line feed is performed automatically when printing reaches the extreme right of the paper.

SEE: PRINT

SAMPLE PROGRAM:

10 LPRINT
20 FOR I = 1 TO 14 : LPRINT "*" : NEXT I
30 LPRINT
40 END

Outputs a series of 14 asterisks to printer.
OPEN

**PURPOSE:**
Declares a file open for use.

**FORMAT:**
OPEN "file descriptor" [ FOR \{ INPUT \} OUTPUT \} AS[#] file number
 Numeric expression

**SAMPLE:**
OPEN "DATA1" FOR OUTPUT AS #1

**PARAMETERS:**
1. file descriptor: String expression
2. file number: Numeric expression truncated to an integer in the range of 1 ≤ file number < 2

**EXPLANATION:**
- The file specified by the file descriptor as the specified file number. Subsequent output to and output from open files is performed by designating the file numbers.
- S0: is the default option when the device name is omitted from the file descriptor.
- Specifying FOR INPUT makes sequential file input possible.
- Specifying FOR OUTPUT makes sequential file output possible. A new file is created on cassette tape.
- The following two conditions are specified when either FOR INPUT or FOR OUTPUT is specified:
  - Cassette tape (CAS0: , CAS1: )
  - Error generated
  - Communications circuit (COM0: )
  - Sequential file input/output specified
- Only one file (#1) can be open at any given time. Attempting to open two or more files results in an OP error.
- Attempting to open a file which is already open results in an OP error.
- File buffer is automatically retained within the stack area. An OM error is generated whenever the stack area becomes full.
- A command can only be executed within a program.

CLOSE

**SAMPLE PROGRAMS:**

1) 10 OPEN "CAS0 : TEST" FOR OUTPUT AS #1
    20 PRINT #1, "WRITE TEST"
    30 CLOSE

Creates sequential file on cassette tape under filename "TEST".

2) 10 OPEN "CAS0 : TEST" FOR INPUT AS #1
    20 INPUT #1, AS
    30 CLOSE

Reads sequential file created in SAMPLE 1.
CLOSE

PURPOSE: Closes files and declares an end to the use of the I/O (input/output) buffer.

FORMAT: CLOSE

EXAMPLE: CLOSE

EXPLANATION:
1. Closes a file and clears the file buffer.
2. An error is not generated even if a file is not open when this command is executed.

SEE: OPEN

SAMPLE PROGRAM:

10 OPEN "CASO : TEST" FOR INPUT AS #1
20 INPUT #1, A$: PRINT A$;
30 IF EOF #1) = 0 THEN 20
40 CLOSE

Reads and displays data from sequential file TEST (stored on cassette tape) until all data have been read.
PRINT 

PURPOSE: Outputs data to a sequential file.

FORMAT: PRINT # [file number, output data [TAB [String expression Numeric expression]]]*

SAMPLE: PRINT #1, AS

PARAMETERS: file number: Numeric expression truncated to an integer in the range of 1 ≤ file number < 2

EXPLANATION:
Sequentially outputs data to the sequential file specified by the file number.
The contents of the output data are the same as those output to the printer by the LPRINT statement (see LPRINT, PRINT).
CR-LF (0DH, 0AH) is output following the last output data when a semicolon and comma are not included.
This statement is valid for sequential files opened for output (FOR OUTPUT), and for communication circuit (COM0:) input/output files.

INPUT #, PRINT, LPRINT

SAMPLE PROGRAMS:

1) 10 OPEN "CAS0 : TEST" FOR OUTPUT AS #1
20 INPUT "DATA =", A$  
30 IF A$ = " " THEN 60  
40 PRINT #1, A$  
50 GOTO 20  
60 CLOSE : END

Creates sequential file on cassette tape for input of characters from keyboard.

2) 10 OPEN "CAS0 : TEST" FOR INPUT AS #1
20 INPUT #1, A$  
30 CLOSE

Reads sequential file on cassette tape created in SAMPLE PROGRAM 1.
PURPOSE: Reads data from a sequential file.

FORMAT:  INPUT # file number , variable name [, variable name]*
          Numeric expression

EXAMPLE: INPUT #1, A

PARAMETERS: file number: Numeric expression truncated to an integer in the range of 1 ≤ file number < 2

EXPLANATION:
1. Reads data from the file specified by the file number.
2. Data are input in the same format as data input using the INPUT statement (see INPUT).
   Consequently, data are delimited using commas, quotation marks, CR codes (0Dh) or CR,
   LF codes (0Dh, 0Ah). Internal codes 00h through 1Fh and 7Fh cannot be input, and leading
   spaces (spaces preceding that data) are disregarded.
3. This statement is valid for sequential files opened for input (FOR INPUT), and for communica-
   tion circuit (COM0:) input/output files.
4. Spaces can also be used as delimiters when data are read to numeric variables.
5. An ST error is generated when data read exceeds 256 characters. Execution continued
   using an ON ERROR statement begins from the 257th character.

SAMPLE PROGRAM:

10 OPEN "CAS0 : TEST" FOR INPUT AS #1
20 INPUT #1, A$  
30 PRINT A$;   
50 IF EOF (1) = 0 THEN 20
20國家 CLOSE : END

Reads and displays data in a sequential file on cassette tape until no more data remain.
**INPUT$**

**PURPOSE:** Reads the specified number of characters from a sequential file.

**FORMAT:**

\[
\text{INPUT$ \ (\ \text{number of characters} \ , \ \# \ \text{file number})} \quad \text{Numeric expression} \quad \text{Numeric expression}
\]

**EXAMPLE:**

\[
\text{INPUT$ (16, \ #1) }
\]

**PARAMETERS:**

1. number of characters: Numeric expression truncated to an integer in the range of \(0 \leq \text{number of characters} < 256\)
2. file number: Numeric expression truncated to an integer in the range of \(1 \leq \text{file number} < 2\)

**EXPLANATION:**

1. Reads the specified number of characters from a sequential file.
2. All codes (00H – FFH) are read as they are.
3. This statement is valid for sequential files opened for input (FOR INPUT), and for communication circuit (COM0:) input/output files.

**SAMPLE PROGRAM:**

```plaintext
10 OPEN "CAS0 : TEST" FOR INPUT AS #1
20 CH$ = INPUT$ (5, #1)
30 CLOSE
40 PRINT CH$
```

Reads and displays first five characters in a sequential file on cassette tape.
PURPOSE: Indicates the end of file reading.

FORMAT: EOF ( file number )
         Numeric expression

EXAMPLE: IF EOF (1) THEN END

PARAMETERS: file number: Numeric expression truncated to an integer in the range of 1 ≤ file number < 2

EXPLANATION:
1. Indicates the end of reading for the file specified by the file number. Generally, this function is assigned a value of 0, but the value becomes -1 when the last record of a file is read.
2. A value of -1 is returned when the receive buffer (for RS-232C applications) becomes empty.
3. This statement is valid for sequential files opened for input (FOR INPUT), and for communications circuit (COM0:) input/output files.
4. Generally, a 0 is returned for sequential files opened for output (FOR OUTPUT).

SAMPLE PROGRAM:

```
10 OPEN "CAS0 : TEST" FOR INPUT AS #1
20 INPUT #1, A$
30 PRINT A$
40 IF EOF (0) THEN 20
50 CLOSE
60 END
```

Reads and displays data in sequential file on cassette tape until no more data remain.
SAVE, SAVE ALL

PURPOSE: Saves a program to a specified file.

FORMAT: SAVE [ALL] "file descriptor" [, A ]
         String expression

EXAMPLE: SAVE "DEMO1"

PARAMETERS:
1. ALL: Outputs all programs from P0 through P9. Can only be specified for output to cassette tape.
2. file descriptor: String expression
3. , A: Specifies ASCII format. Binary internal format is the default option when omitted. Cannot be specified while SAVE ALL is specified.

EXPLANATION:
1. Outputs the currently specified program area contents to the file specified by the file descriptor.
2. Specifying ALL outputs programs from areas P0 through P9 to cassette tape as an ALL file.
3. CAS0: is the default option when the device name is omitted from the file descriptor. When the entire file descriptor is omitted, the file is output to cassette tape and saved without a filename.
4. Specifying "", A" causes the program to be converted to and saved in ASCII format. This format uses alphabetic characters such as those which appear when the LIST command is executed.
5. Data are output as they are in binary format when "", A" is not specified. However, files are saved in ASCII format whenever COM0: is specified in the file descriptor, regardless of the "", A" specification.
6. Files for which a password has been registered cannot be saved in ASCII format.
7. This command causes all open files to be closed and enters command input standby once the SAVE execution is complete.
8. This command cannot be executed while program execution is halted ("STOP" displayed).
9. This command cannot be executed in the CAL mode.

SEE: LOAD, PASS, LOAD ALL

SAMPLE EXECUTION: SAVE "CAS0 : TEST"

Saves a program on cassette tape under filename TEST.
LOAD, LOAD ALL

PURPOSE: Reads from a file into memory.

FORMAT: LOAD [ALL] “file descriptor” [, A]
         String expression

EXAMPLE: LOAD “DEMO1”

PARAMETERS: 1. ALL: Inputs programs to program areas P0 through P9. Can only be
              specified for input from cassette tape.
2. file descriptor: String expression
3. , A: Specifies ASCII format for cassette tape. Binary format is the
       default option when , A is omitted. ASCII format is the default option
       for the communications circuit, whether specified or not.

EXPLANATION:
1. Reads from the file specified by the file descriptor to the currently specified program area.
   The format of the file can be either internal or ASCII format.
2. CAS0: is the default option when the device name is omitted from the file descriptor.
3. Files already in existence before execution of this command are deleted, and the specified files are loaded in their place.
4. This command closes all open files and the computer waits for command input once load is complete.
5. Passwords and program loading.

<table>
<thead>
<tr>
<th>Computer</th>
<th>Loaded program</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>Password</td>
<td>LOAD possible when passwords are identical only</td>
</tr>
<tr>
<td>Password</td>
<td>No password</td>
<td>LOAD possible</td>
</tr>
<tr>
<td>No password</td>
<td>No password</td>
<td>LOAD possible</td>
</tr>
<tr>
<td>No password</td>
<td>Password</td>
<td>LOAD possible (under program password)</td>
</tr>
</tbody>
</table>

6. Specifying ALL reads ALL files (files with attribute A, created using SAVE ALL) from cassette tape into areas P0 through P9.
7. This command cannot be executed in the CAL mode.
8. This command cannot be executed while program execution is halted.
9. The first file on the cassette tape with an attribute which matches the one specified is the default option when the entire file descriptor is omitted.

LOAD: first file saved in internal format (attribute B)
LOAD ALL: first file saved in ALL format (attribute A)
LOAD , A: first file saved in ASCII format (attribute S)

SEE: SAVE

SAMPLE EXECUTION: LOAD ""CAS0 : TEST"

Reads program under filename TEST from cassette tape.
VERIFY

PURPOSE: Verifies the contents of a file stored on cassette tape.

FORMAT: VERIFY "file descriptor"
        String expression

EXAMPLE: VERIFY "CAS0: DEMO"

PARAMETERS: file descriptor: String expression

EXPLANATION:
Verifies the contents of a file stored on cassette tape.
Parity and checksum data included within the file itself are used for checking.
This command cannot be executed in the CAL mode.
This command closes all open files.
The first program found is checked when the filename is omitted.
This command cannot be executed while program execution is halted ("STOP" displayed).

SEE: SAVE, LOAD

SAMPLE EXECUTION: VERIFY "CAS0 : TEST"

Confirms whether or not program on cassette tape has been correctly stored under filename TEST.
DATA BANK COMMANDS

NEW 

PURPOSE: Clears DATA BANK data.
EXPLANATION:
1. Clears all data stored under the DATA BANK function.
2. This command cannot be executed for data protected by a password.
3. This command cannot be executed in the CAL mode, but in the BASIC mode.
SAMPLE EXECUTION: NEW# EXEC
Clears DATA BANK data.

LIST 

PURPOSE: Displays all DATA BANK data.
EXPLANATION:
1. Displays in record sequence all data stored under the DATA BANK function.
2. The display shows the record number and DATA BANK data.
3. The listing can be halted at any time by pressing EXEC, and resumed by pressing any key other than EXEC and CLS.
4. The listing can also be halted at any time by pressing QUIT.
5. This command cannot be executed for data protected by a password.
6. This command cannot be executed in the CAL mode, but in the BASIC mode.
SEE: LLIST#
SAMPLE EXECUTION: LIST# EXEC
Lists DATA BANK data on display.
LLIST #

PURPOSE: Outputs all DATA BANK data to printer.

EXPLANATION:
1. Outputs to the printer in record sequence all data stored under the DATA BANK function.
2. The record number and DATA BANK data are both printed.
3. This command cannot be executed for data protected by a password.
4. This command cannot be executed in the CAL mode, but in the BASIC mode.

SEE: LIST#

SAMPLE EXECUTION: LLIST # EX

Outputs DATA BANK data to printer.

SAVE #

PURPOSE: Outputs DATA BANK data to file specified by file descriptor.

FORMAT: SAVE# [ file descriptor ].

EXAMPLE: SAVE# "CAS0 : TEST"

PARAMETERS: file descriptor: String expression

EXPLANATION:
1. Outputs DATA BANK data to a file specified by the file descriptor.
2. Data are output in ASCII format.
3. CAS0: is the default option when the device name is omitted from the file descriptor.
4. When the entire file descriptor is omitted, the file is output to cassette tape and saved without a filename.
5. This command cannot be executed in the CAL mode, but in the BASIC mode.
6. This command cannot be executed while program execution is halted ("STOP" displayed).
7. This command cannot be executed for data protected by a password.

SEE: LOAD#

SAMPLE EXECUTION: SAVE# "CAS0 : TEL" EX

Save DATA BANK data to cassette tape under filename TEL.
LOAD #

PURPOSE: Reads data into DATA BANK area.

FORMAT: LOAD # \( \left[ \text{file descriptor} \right] \) \([, \ M]\) \(\text{String expression}\)

EXAMPLE: LOAD # "CAS0 : TEST"

PARAMETERS: 1. file descriptor: String expression
2. , M: Indicates that current execution is append to existing data.

EXPLANATION:
1. Reads data to the DATA BANK area from the file specified by the file descriptor.
2. The current contents of the DATA BANK area are deleted when "", M"" is not specified.
   Specifying "", M"" indicates that the new data are to be appended to the end of the current contents of the DATA BANK area.
3. CAS0: is the default option when the device name is omitted from the file descriptor.
4. The first file on the cassette tape with an attribute (S) which matches the one specified is the default option when the entire file descriptor is omitted.
5. This command cannot be executed in the CAL mode, but in the BASIC mode.

SAMPLE EXECUTION: LOAD # "CAS0 : TEL2", M

Reads memo data file stored on cassette tape under filename TEL2 and appends to current DATA BANK area contents.
**READ #**

**PURPOSE:** Reads data from DATA BANK area.

**FORMAT:** READ # variable name [, variable name ]*

**EXAMPLE:** READ # A$, X

**PARAMETERS:** variable name

**EXPLANATION:**
1. Sequentially reads data stored in the DATA BANK area and assigns them to variables.
2. Numeric data can only be read into numeric variables, and string data only into string variables. Mismatching data and variables generates an error.
3. Data items can be delimited by commas.
4. A DA error is generated when data are not present to be read.
5. The read sequence can be altered using the RESTORE # command.
6. Spaces in front of data items are skipped.
7. Data included within quotation marks are read as a single string.
8. This command cannot be executed in the CAL mode, but in the BASIC mode.

**SEE:** RESTORE #, WRITE #

**SAMPLE PROGRAM:**

```
10   RESTORE # "RED", 0, 50
20   READ # A$
30   PRINT A$
40   GOTO 10
50   PRINT "NO DATA!"
```

Searches and displays data items which start with "RED" within DATA BANK area. Message "NO DATA!" appears when such data items are not found.
**PURPOSE:** Searches specific data in the DATA BANK area and changes the read sequence of DATA BANK data.

**FORMAT:**

```
RESTORE # [ "object string" [ String expression [ ], [ 1 ] [ line number [ program area number ] ] ] ]
```

**EXAMPLE:**

```
RESTORE # "SMITH"
```

**PARAMETERS:**
1. **object string:** String expression
2. **line number:** Numeric expression. Integer within the range of 0 < line number ≤ 65536
3. **program area number:** Numeric expression. Integer within the range of 0 ≤ program area number < 10

**EXPLANATION:**

1. Searches specific data in the DATA BANK area and sets the DATA BANK area pointer position. Subsequent executions of the READ # and WRITE # commands are performed from the new pointer position.
2. The relationship between the parameters and the object string are as follows:
   i) RESTORE #
      Omitting all parameters sets the DATA BANK area pointer to the beginning of the data to be read by the next READ # command.
   ii) RESTORE # "object string"
      Sets the DATA BANK area pointer to the position of the specified object string. Strings are delimited by commas, and not by spaces. A DA error is generated when the object string cannot be found.
   iii) RESTORE # "object string", { 0, 1 }
        0: Same as ii above.
        1: The first data of the record (line) that includes object string is read by the following READ # statement.
   iv) RESTORE # "object string" [ , { 0, 1 } ] [ line number [ program area number ] ]
      Execution branches to specified line or program area when the object string is not found.

* Search is conducted from the present pointer position forward to the higher record number. The following procedure is used to search from the beginning of entire data:

```
RESTORE # : RESTORE # "object string"
```

**SAMPLE PROGRAM:**

```
10 RESTORE # "YOU", 0, 50
20 READ # A$
30 PRINT A$
40 GOTO 10
50 PRINT "NO DATA!"
```

Searches for data beginning with "YOU", and displays "NO DATA!" if not found.
**WRITE #**

**PURPOSE:** Rewrites and deletes DATA BANK data.

**FORMAT:**

\[
\text{WRITE } # \left( \frac{\text{DATA BANK data}}{\text{Expression}} \right) \left( \frac{\text{DATA BANK data}}{\text{Expression}} \right)
\]

**EXAMPLE:**

WRITE # "ABCDEF"

**PARAMETERS:**

DATA BANK data: String or numeric expression

**EXPLANATION:**

1. Sequentially writes DATA BANK data from the current DATA BANK area pointer (see RESTORE #).
2. New data are written regardless of whether or not data already exist at the pointer location.
3. The entire record (line) is cleared when this command is executed without any DATA BANK data expressions.
4. Multiple data items can be delimited using commas.
5. The DATA BANK area pointer stays at the next data item written after execution of this command. Further data item writing begins from this point unless the pointer position is changed using RESTORE #.
6. 255 characters per line can be written using this command, and an error is generated when this limit is exceeded.
7. Numeric expressions written using this command are written using the same format as PRINT statement display. Note, however, that the SET statement has no effect here.
8. This command cannot be used to write character codes &H1F or lower.

**SAMPLE PROGRAM:**

```
10 RESTORE #
20 RESTORE # "YOU", 0, 50
30 WRITE # "SHE"
40 GOTO 20
50 PRINT "COMPLETE!"
60 END
```

Changes DATA BANK data beginning with "YOU" to "SHE".
11-1 LIBRARY EXECUTION

11-1-1 Activating The Library

The library function of the FX-850P provides a total of 116 different utilities divided into a mathematical library, a statistical library, and physics and scientific library. The two methods described below can be used to activate the desired library in the CAL mode.

1. Library number + \text{on} key

Activation of the library using this method is achieved by first entering a library number and then pressing the \text{on} key.

\textbf{EXAMPLE}

Activation of the library utility for solution of a quadratic equation (Library Number 5050).

\begin{center}
\begin{tabular}{|c|}
\hline
5050 \text{on} \\
\hline
\end{tabular}
\end{center}

\begin{center}
\begin{tabular}{|c|}
\hline
ax^2 + bx + c = 0 \\
\hline
a = 1 \\
\hline
\end{tabular}
\end{center}

* The cursor moves to the next line with no further operation when an invalid library number is entered.

One of the following two operations is performed when the \text{on} key is pressed without entry of a library number.

i) Pressing \text{on} immediately after power is switched ON

\begin{center}
\begin{tabular}{|c|}
\hline
\text{on} \\
\hline
\end{tabular}
\end{center}

\begin{center}
\begin{tabular}{|c|}
\hline
Prime factors \\
Base ?? \\
\hline
\end{tabular}
\end{center}

(Switch power ON)

This operation activates the prime factor analysis library utility (Library Number 5010).

ii) Pressing \text{on} after execution of a library utility

\begin{center}
\begin{tabular}{|c|}
\hline
\text{on} \\
\hline
\end{tabular}
\end{center}

\begin{center}
\begin{tabular}{|c|}
\hline
Break \\
\hline
\end{tabular}
\end{center}

\begin{center}
\begin{tabular}{|c|}
\hline
y=ax+b = (x1,y1),(x2,y2) \\
x1=0 ?? \\
\hline
\end{tabular}
\end{center}

(Press to break)

(Press \text{on})
WRITE #

PURPOSE: Rewrites and deletes DATA BANK data.

FORMAT: \[
\text{WRITE} # \left\{ \frac{\text{DATA BANK data}}{\text{Expression}}, \frac{\text{DATA BANK data}}{\text{Expression}} \right\}
\]

EXAMPLE: WRITE # "ABCDEF"

PARAMETERS: DATA BANK data: String or numeric expression

EXPLANATION:
1. Sequentially writes DATA BANK data from the current DATA BANK area pointer (see RESTORE #).
2. New data are written regardless of whether or not data already exist at the pointer location.
3. The entire record (line) is cleared when this command is executed without any DATA BANK data expressions.
4. Multiple data items can be delimited using commas.
5. The DATA BANK area pointer stays at the next data item written after execution of this command. Further data item writing begins from this point unless the pointer position is changed using RESTORE #.
6. 255 characters per line can be written using this command, and an error is generated when this limit is exceeded.
7. Numeric expressions written using this command are written using the same format as PRINT statement display. Note, however, that the SET statement has no effect here.
8. This command cannot be used to write character codes &H1F or lower.

SAMPLE PROGRAM:

```
10 RESTORE #
20 RESTORE # "YOU", 0, 50
30 WRITE # "SHE"
40 GOTO 20
50 PRINT "COMPLETE!"
60 END
```

Changes DATA BANK data beginning with "YOU" to "SHE".
11-1 LIBRARY EXECUTION

11-1-1 Activating The Library

The library function of the FX-850P provides a total of 116 different utilities divided into a mathematical library, a statistical library, and physics and scientific library. The two methods described below can be used to activate the desired library in the CAL mode.

1. Library number + [LIB] key

Activation of the library using this method is achieved by first entering a library number and then pressing the [LIB] key.

EXAMPLE

Activation of the library utility for solution of a quadratic equation (Library Number 5050).

\[ a \times x^2 + b \times x + c = 0 \]

\[ a = 1 \quad \text{(Library number entry)} \]

* The cursor moves to the next line with no further operation when an invalid library number is entered.

One of the following two operations is performed when the [LIB] key is pressed without entry of a library number.

i) Pressing [LIB] immediately after power is switched ON

\[ \text{Prime factors} \quad (2 \leq \text{Base} < 10^{10}) \]

\[ \text{Base} \quad ? \]

(Switch power ON)

(Press [LIB])

This operation activates the prime factor analysis library utility (Library Number 5010).

ii) Pressing [LIB] after execution of a library utility

\[ y = ax + b \quad \text{--} \quad (x_1, y_1), (x_2, y_2) \]

\[ x_1 = 0 \quad ? \]

(Press to break)

(Press [LIB])
In this case, the previous library utility (here, Library Number 5510; STRAIGHT LINE PASSING THROUGH TWO POINTS) is reactivated.

* In this example, the [A] key was pressed immediately following [AD]. The same result is produced when manual calculations or a BASIC program is executed following [AD].

2. Selection using the [A] key

Pressing the [A] key produces a display of the library utilities built into the FX-850P. The following operations can be used to locate a specific utility.

i) [6] and [7] keys are used to respectively scroll up and down through the utility list. Holding either key down results in high speed display change.

[6] displays the first library utility (memory calculation, Library Number 1000).
[7] displays the last library utility (ratio difference test, Library Number 6772).
[8] displays following library utility contents.

ii) Pressing either [6] or [7] when the desired library utility is displayed executes the utility.

**EXAMPLE 1**

Activation of numerical solution (method of bisection) after power is switched ON.

- (Switch power ON)

Base = a b c x ...


5090: Numerical solution f(x) = 0
Method of bisection

[8] Method of bisection f(x) = 0
\[ f(x) \neq 0 \] \[ \text{ at } x \] \[ \text{ at } x \] \[ \text{ at } x \] \[ \text{ at } x \] \[ \text{ at } x \] \[ \text{ at } x \]


[8] (Execution using [8])

**EXAMPLE 2**

Termination of bisection method in EXAMPLE 1 and activation of quadratic equation solution utility.

[6] Break

[6] 5090: Numerical solution f(x) = 0
Method of bisection

[6] (Termination of library utility)

[6] 5050: ax^2 + bx + c = 0

[6] (Library list display)


5050: ax^2 + bx + c = 0


[8] (Activation of library utility)

11-1-2 Library Termination

Execution of a library utility can be terminated by pressing the [A] key.

ax^2 + bx + c = 0
a = 1

[6] Break

[6] (Termination of library utility)
11-1-3 Library Activation Display

The displays that appear immediately following activation of the library are of two types, and are referred to throughout this manual as follows.

1. Initial display
Display immediately following library activation for value input, YES/NO selection, or list display.

**EXAMPLE 1**
Immediately following activation of prime factor analysis library utility (Library Number 5010).

\[
\text{Prime factors} \quad (2 \leq \text{Base} \leq 10)^9
\]

**EXAMPLE 2**
Immediately following activation of interval estimation library utility (Library Number 6610).

\[
N(\mu, \sigma^2) \quad a < \mu < b \quad \sigma^2 \text{ known} \\
\text{Input new data (Y/N)?}
\]

**EXAMPLE 3**
Immediately following activation of formula library utility (Library Number 5800).

\[
a^2 - b^2 = (a + b)(a - b)
\]

2. Menu display
Display immediately following library activation for process selection.

**EXAMPLE 1**
Immediately following activation of Newton's method library utility (Library Number 5080).

\[
\text{Newton's method} \\
1: f(x) \neq 0 \quad 2: h, f(x) \neq 0
\]

**EXAMPLE 2**
Immediately following activation of matrix operation library function (Library Number 5100).

\[
\text{Matrix} \quad A(2,2) : B(2,2) \\
\text{A, B, D, I, T, K, +, -, x, M, L, C, P?}
\]
11-1-4 Examples Used in This Manual

The examples shown in this manual are generally presented as being performed immediately following library activation. When the library is activated, certain values (0 or 1) are stored for the variables used within the library. Continuously using the library without a break causes the values which have been entered or calculated to be retained. When inputting data, the values assigned to variables are displayed as shown in the display illustrated below (actual display differs according to the library utility being used).

\[
\begin{array}{c}
\text{Ratio interval estimation; Library Number} \\
6670
\end{array}
\]

At this time, the \text{ENT} key can be pressed without changing the displayed value, or the displayed value can be changed before pressing \text{ENT}.

11-1-5 Precautions When Using the Library

1. Library executions can be performed in the CAL mode only.
2. A number of different types of variables are used in library calculations. Using a large number of variables in various library utilities may cause library execution speed to decrease. Speed can be increased, in this case, by executing the CLEAR statement before activating the library function. It should be noted, however, that the CLEAR statement clears all variables currently stored in memory.
3. Activation of the library automatically switches the PRINT mode OFF and executes the DIM command. This means that a PRINT mode ON specification or DEFM command executed before the library is activated is canceled.
4. Numeric values used during library executions should have mantissas of 10 digits or fewer.
5. Library variable names consist of single lower case alphabetic characters (a – z). Statistical variable names in the library are preceded by the letter "s" (sa – sz).
6. All library utilities are created using the BASIC language.
7. The \text{ENT} symbol is shown on the display during library execution and list display.
8. The \text{ENT} can be pressed while the previously entered data are displayed during data input to enter the displayed data again.

**EXAMPLE**

\[
\begin{array}{c}
\text{test} \\
\text{H₀: } p = 0 \Rightarrow p \\
\text{H₁: } p > 0
\end{array}
\]

Pressing \text{ENT} here inputs 0.02 as the value for \(P₀\).

Execution of certain library utilities automatically switches to the lower case mode or the RAD angle unit (from DEG) mode. Since pressing the \text{ENT} key terminates execution while retaining the lower case or angle unit mode, the mode automatically switched to should be manually changed as required before execution of another library utility or calculation.

- Library utilities automatically switching to lower case mode
  5080, 5090, 5200, 5220
- Library utilities automatically switching angle unit to RAD
  5080, 5090, 5200, 5220, 5625, 6230, 6240, 6430, 6440, 6450, 6620, 6650, 6660, 6720, 6721, 6722, 6740, 6741, 6742, 6750, 6751, 6752
This function makes it possible to use the cursor keys to perform the four key memory (MC, MR, M−, M+) operations.

The following list shows the corresponding memory operation that corresponds to each cursor key.

- **MC**: (Memory Clear) Clears data stored in memory
- **MR**: (Memory Recall) Recalls data stored in memory
- **M−**: (Memory minus) Subtracts from memory
- **M+**: (Memory plus) Adds to memory

Both the calculation result and memory contents are simultaneously shown at the bottom of the display.

<table>
<thead>
<tr>
<th>MC(1)</th>
<th>MR(1)</th>
<th>M−(−)</th>
<th>M+(−)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Values can be corrected using the * key (1-character delete) or ** key (all value clear).

Besides the four basic arithmetic functions, numeric scientific function, logical operation, and comparison calculations can all be performed. One-key commands, however, cannot be used for numeric scientific function calculations, and direct function keys cannot be used.

**EXAMPLE**

**cannot be used to enter \( \sin 30^\circ \). It must be entered as **. 

The formula memory is used for memory calculations. Therefore, it should be noted that contents of the formula memory are changed when memory calculations are performed.

**OPERATION**

<table>
<thead>
<tr>
<th>MC(1)</th>
<th>MR(1)</th>
<th>M−(−)</th>
<th>M+(−)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**EXAMPLE 1**

Perform the calculation: \( 15 \div 3 + 7 - 6 = 6 \)

<table>
<thead>
<tr>
<th>MC(1)</th>
<th>MR(1)</th>
<th>M−(−)</th>
<th>M+(−)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

15 \( \div 3 + 7 - 6 = 6 \)

<table>
<thead>
<tr>
<th>MC(1)</th>
<th>MR(1)</th>
<th>M−(−)</th>
<th>M+(−)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

15 \( \div 3 + 7 - 6 \) (Formula input)

<table>
<thead>
<tr>
<th>MC(1)</th>
<th>MR(1)</th>
<th>M−(−)</th>
<th>M+(−)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

15 \( \div 3 + 7 - 6 \) (Formula execution)
EXAMPLE 2
Perform the following calculations: \( 120 \times 1.4 = 168 \)
\( 1.4 \times 170 = 238 \)

<table>
<thead>
<tr>
<th>Operation</th>
<th>Display</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>120 ( \times ) 1.4</td>
<td>( 120 \times 1.4 )</td>
<td>Formula input</td>
</tr>
<tr>
<td>M+</td>
<td>168</td>
<td>Memory execution</td>
</tr>
<tr>
<td>1.4 ( \times ) 170</td>
<td>( 1.4 \times 170 )</td>
<td>Formula execution</td>
</tr>
<tr>
<td>M+</td>
<td>238</td>
<td>Memory execution</td>
</tr>
</tbody>
</table>

EXAMPLE 3
Perform the following calculation: \( 3 + 7 + \sin 30^\circ \) (angle unit = degree)

<table>
<thead>
<tr>
<th>Operation</th>
<th>Display</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>Storage of value in memory</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>Add to memory</td>
</tr>
<tr>
<td>SIN30</td>
<td>( \sin 30^\circ )</td>
<td>Add to memory following function calculation</td>
</tr>
</tbody>
</table>

Set the mode for the desired angle unit (DEG, RAD, GRA) before activating the library.
Performs prime factor analysis on an input value base. The input range of entered value a is an integer within the range of \(2 \leq a < 10^{10}\). The analysis is performed by first determining if the value input for a is divisible by 2 or by b, which is assigned sequential odd numbers (3, 5, 7...).

When b is a prime factor, the formula \(a_i = \frac{a_i - 4}{b}\) is applied and division is repeated until \(\sqrt{a_i} + 1 \leq b\).

**OPERATION**

**5010**

<table>
<thead>
<tr>
<th>Prime factors (2 ≤ Base &lt; 10^10)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base ?</strong></td>
</tr>
</tbody>
</table>

**EXAMPLE**

Perform prime factor analysis for a base of 100.

<table>
<thead>
<tr>
<th>Prime factors (2 ≤ Base &lt; 10^10)</th>
<th>(Base input)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base ? 100</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prime factors (2 ≤ Base &lt; 10^10)</th>
<th>(Calculation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 = 2 × 2 × 5 × 2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prime factors (2 ≤ Base &lt; 10^10)</th>
<th>(Result display)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base 7</td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

Here, the result of the prime factor analysis is \(100 = 2^2 \times 5^2\).
## GREATEST COMMON MEASURE/LEAST COMMON MULTIPLE

Determines the greatest common measure (GCM) and least common multiple (LCM) for two entered integers \((a, b)\), within the range of \(1 \leq a < 10^{10}, 1 \leq b < 10^{10}\). The GCM and LCM are determined using the Euclidean method.

### OPERATION

**5020**

\[
\begin{align*}
\text{G.C.M.} & \quad \text{&} \quad \text{L.C.M.} \quad (1 \leq a, b < 10^{10}) \\
\text{a} & = 1 \quad ?_-
\end{align*}
\]

**EXAMPLE**

Determine the GCM and LCM when \(a = 5\) and \(b = 2\).

\[
\begin{align*}
\text{G.C.M.} & \quad \text{&} \quad \text{L.C.M.} \quad (1 \leq a, b < 10^{10}) \\
\text{a} & = 5 \quad ?_-
\end{align*}
\]

5

\[
\text{G.C.M.} \quad \text{&} \quad \text{L.C.M.} \quad (1 \leq a, b < 10^{10})
\]

(2) Value a input

\[
\begin{align*}
\text{G.C.M.} & = 1 \\
\text{L.C.M.} & = 10
\end{align*}
\]

(2) Value b input

\[
\text{G.C.M.} \quad \text{&} \quad \text{L.C.M.} \quad (1 \leq a, b < 10^{10})
\]

(2) Return to initial display

Here, the GCM of 2 and 5 is 1, while the LCM is 10.

## SIMULTANEOUS EQUATIONS

(GAUSS-JORDAN ELIMINATION)

Solves for \(x_1 - x_n\) in the following \(n\) simultaneous equations \((2 \leq n \leq 7)\) for input of coefficients \(a_1 - a_n, b_1 - b_n\) ... and \(y_1 - y_n\).

\[
\begin{align*}
a_1 \cdot x_1 + b_1 \cdot x_2 + c_1 \cdot x_3 + \cdots &= y_1 \\
a_2 \cdot x_2 + b_2 \cdot x_2 + c_2 \cdot x_3 + \cdots &= y_2 \\
&\vdots \\
a_n \cdot x_n + b_n \cdot x_n + c_n \cdot x_n + \cdots &= y_n
\end{align*}
\]

Solutions may not be exact for coefficients with a difference in excess of \(1 \times 10^{10}\) due to internal rounding.

### OPERATION

**5040**

\[
\begin{align*}
\text{Solve} \quad b_1 x_1 & + b_2 x_2 + c x_3 + \cdots = y \\
\text{for} \quad n & \leq 7
\end{align*}
\]

Pressing \(\text{ESC}\) during coefficient input returns to the previous coefficient entry.

Pressing \(\times\) or \(\text{ESC}\) during display of a solution scrolls to the following solution, while \(\text{ESC}\) scrolls to the previous solution.

The message "not found" appears on the display when a solution cannot be found.
EXAMPLE

Solve the following simultaneous cubic equations for $x_1$, $x_2$, and $x_3$.

$2x_1 + 3x_2 - x_3 = 15$
$3x_1 - 2x_2 + 2x_3 = 4$
$5x_1 + 3x_2 - 4x_3 = 9$

Here, the solutions of the simultaneous equations are $x_1 = 2$, $x_2 = 5$, $x_3 = 4$. 
Determines the solution for \( \alpha \) and \( \beta \) when coefficients \( a \), \( b \), and \( c \) are input for the quadratic equation \( ax^2 + bx + c = 0 \).

Root equations are used to determine the solution.

Root equation: \( x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \)

When \( d = b^2 - 4ac \):

i) When \( d > 0 \), two real roots (\( \alpha \), \( \beta \)) are present: \( \alpha = \frac{-b + \sqrt{d}}{2a} \), \( \beta = \frac{-b - \sqrt{d}}{2a} \)

ii) When \( d = 0 \), one real root (\( \alpha \)) is present: \( \alpha = \frac{-b}{2a} \) (multiple root)

iii) When \( d < 0 \), two imaginary roots (\( \alpha \), \( \beta \)) are present:

\[
\alpha = \frac{-b}{2a} + \frac{\sqrt{-d}}{2a} \quad i, \quad \beta = \frac{-b}{2a} - \frac{\sqrt{-d}}{2a} \quad i
\]

**OPERATION**

**EXAMPLE**

Determine the solution of the following quadratic equation:

\( 2x^2 - 5x + 3 = 0 \)

Here, the solutions of \( 2x^2 - 5x + 3 = 0 \) are \( \alpha = 1.5 \), \( \beta = 1 \).

**SOLUTION DISPLAY**

Pressing \( \alpha \) or \( \beta \) scrolls from display of \( \alpha \) to \( \beta \) (only \( \alpha \) displayed for multiple root). Pressing \( \alpha \) while \( \beta \) is displayed returns to the display of \( \alpha \).
CUBIC EQUATIONS

Determines the solution for $\alpha$, $\beta$ and $\gamma$ when coefficients $a$, $b$, $c$, and $d$ are input for the cubic equation $ax^3 + bx^2 + cx + d = 0$.

Root equations are used to determine the solution.

Transformation to $y^3 + 3py + q = 0$ can be performed

when $x = y - \frac{b}{3a}$, $p = \frac{c - b^2}{9a^2}$, $q = \frac{2b^3}{27a^3} - \frac{bc}{3a^2} + \frac{b}{a}$ are substituted in $ax^3 + bx^2 + cx + d = 0$.

Here, substituting $A = \frac{3\sqrt{q + \sqrt{c}}}{2}$, $B = \frac{3\sqrt{q - \sqrt{c}}}{2}$, $c = q^2 + 4p^3$ results in the following:

i) When $c > 0$, one real root ($\alpha$), and two imaginary roots ($\beta$, $\gamma$) are present:

$$\alpha = -(A + B), \quad \beta = \frac{A + B}{2} + \frac{\sqrt{3}}{2} (A - B)i, \quad \gamma = \frac{A + B}{2} - \frac{\sqrt{3}}{2} (A - B)i$$

ii) When $c = 0$, $p = 0$, one real root ($\alpha$) is present: $\alpha = -(A + B)$

iii) When $c = 0$, $p \neq 0$, two real roots ($\alpha$, $\beta$) are present:

$$\alpha = -(A + B), \quad \beta = \frac{A + B}{2} \quad \text{(multiple roots)}$$

iv) When $c < 0$, three real roots ($\alpha$, $\beta$, $\gamma$) are present:

$$\alpha = -2\sqrt{-p}\cos\theta, \quad \beta = -2\sqrt{-p}\cos(\theta + 120^\circ), \quad \gamma = -2\sqrt{-p}\cos(\theta + 240^\circ) \quad \theta = \frac{1}{3}\cos^{-1}\frac{q}{2\sqrt{-p^3}}$$

OPERATION

5060 [LB]

5060 [LB]

EXAMPLE

Determine the solution of the following cubic equation:

$$2x^3 + x^2 - 13x + 6 = 0$$

2 [EX]

1 [EX]

-13 [EX]

6 [EX]

$$(\text{Coefficient } a \text{ input})$$

$$(\text{Coefficient } b \text{ input})$$

$$(\text{Coefficient } c \text{ input})$$

$$(\text{Coefficient } d \text{ input})$$

$$s(x-\alpha)(x-\beta)(x-\gamma) = 0$$

$$(\text{Solution } \alpha \text{ display})$$

$$\alpha = -3$$

$$\beta = 2$$

$$(\text{Solution } \beta \text{ display})$$
Here, the solutions of $2x^3 + x^2 - 13x + 6 = 0$ are $\alpha = -3$, $\beta = 2$, $\gamma = 0.5$.

**SOLUTION DISPLAY**

Pressing **E** or **X** displays $\alpha$, $\beta$ and $\gamma$ in sequence. Pressing **E** while $\beta$ or $\gamma$ is displayed returns to the display of $\alpha$ or $\beta$. Only $\alpha$ or $\alpha$ and $\beta$ are displayed in the case of multiple roots.

**CUBIC EQUATION FLOWCHART**
NUMERIC SOLUTION OF AN EQUATION
(NEWTON'S METHOD)

Determines the solution of the function \( y = f(x) \) graphed below for \( f(x) = 0 \), using Newton's Method.

The following parameters are specified in order to determine the numeric solution using Newton's Method.

\( x_0 \) : Initial value
\( h \) : Minute interval for x-axis when performing numerical differentiation at point \((x, f(x))\)
\( f'(x) = \frac{f(x + h) - f(x)}{h} \)
\( \epsilon \) : Solution convergence (\( \epsilon > |x_{n+1} - x_n| \) : continuously calculate and return value of \( \epsilon \) as long as inequality is true)
\( \text{loop} \) : Maximum number of convergences (positive integer)

* The following arithmetic operators and functions can be applied here:
  +, -, *, /, ^, SIN, COS, TAN, ASN, ACS, ATN, LOG, LN, EXP, SQR, HYP
* The variable name for the function \( f(x) \) must be represented by \( x \).
* The value input for \( \epsilon \) must be \( 1 \times 10^{-90} \) or more. Since internal calculations are performed in 12 digits, smaller values have little meaning.

OPERATION

5080 [LIB]

\[ \text{Newton's method } f(x) = 0 \]
\[ 1: f(x), x_0 \]
\[ 2: h, \epsilon, \text{loop} \]

The menu display illustrated above appears when the library is activated. Either 1 or 2 should be selected in accordance with the type of processing to be performed.

1 : Function \( f(x) \) specification/initial value input
2 : Input of minute interval, convergence condition, and maximum number of convergences
### Example

Determine the \( f(x) = 0 \) solution of the following equation for \( f(x) = 2x^2 + 3x^2 - x - 5 \), where the minute interval is 0.00001, the convergence condition is 0.0001, and the maximum number of convergences is 30.

![Graph of the function](image)

<table>
<thead>
<tr>
<th>Newton's method ( f(x) = 0 )</th>
<th>Parameter input selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>( f(x) = (f(x+h) - f(x))/h \ (h &gt; 0) )</td>
</tr>
<tr>
<td>0.00001 ( \text{err} )</td>
<td>( h = 0.000001 ) ?_</td>
</tr>
<tr>
<td>0.0001 ( \text{err} )</td>
<td>( \epsilon = 0.0000001 ) ?_</td>
</tr>
<tr>
<td>30 ( \text{err} )</td>
<td>( \text{Max loop } (n &gt; 0) )</td>
</tr>
<tr>
<td>( \text{err} )</td>
<td>( n = 20 ) ?_</td>
</tr>
<tr>
<td>1</td>
<td>( \text{Define function} \ f(x) ?_ )</td>
</tr>
<tr>
<td>2(x^3 + 3x^2 - x - 5 ) ( \text{err} )</td>
<td>( \text{Function input} )</td>
</tr>
<tr>
<td>1 ( \text{err} )</td>
<td>( f(x) = 0 ) ?_</td>
</tr>
<tr>
<td>( \text{err} )</td>
<td>( f(x) = 2x^3 + 3x^2 - x - 5 )</td>
</tr>
<tr>
<td>( \text{err} )</td>
<td>( x = 0 ) ?_</td>
</tr>
<tr>
<td>( \text{err} )</td>
<td>( f(x) = 2x^3 + 3x^2 - x - 5 )</td>
</tr>
<tr>
<td>( \text{err} )</td>
<td>( x = 1 ) ( \text{err} )</td>
</tr>
</tbody>
</table>

This display indicates that the solution for the example equation is 1.0849.

The message "not found" is displayed when a solution cannot be found.

```
\( f(x) = x^2 + 1 \)
not found
```

Pressing \( \text{err} \) at this time returns the display to point at which calculation was discontinued. Pressing \( \text{err} \) again returns to menu for numeric solution of an equation (certain calculations may not initially display discontinued point display).

```
\( f(x) = x^2 + 1 \)
\( \text{loop} = 20 \)
\( x_n = 1.070479459 \)
```
Determines the solution of the function $y = f(x)$ graphed below for $f(x) = 0$, using the bisection method.

The following parameters are specified in order to determine the numeric solution using the bisection method.

$x_0$, $x_1$ : Initial value
$\epsilon$ : Solution convergence ($\epsilon > |x_{n+1} - x_n|$ : continuously calculate and return value of $\epsilon$ as long as inequality is true)
loop : Maximum number of convergences (positive integer)

* The following arithmetic operators and functions can be applied here:
  +, -, *, /, ^, SIN, COS, TAN, ASN, ACS, ATN, LOG, LN, EXP, SQR, HYP
* The variable name for the function $f(x)$ must be represented by $x$.
* The value input for $\epsilon$ must be $1E-90$ or more. Since internal calculations are performed in 12 digits, smaller values have little meaning.

**OPERATION**

5090 [UB] Method of bisection $f(x) = 0$
1: $f(x), x0, x1$ 2: $\epsilon, loop$

The menu display illustrated above appears when the library is activated. Either 1 or 2 should be selected in accordance with the type of processing to be performed.

1 : Function $f(x)$ specification/initial value input
2 : Input of convergence condition and maximum number of convergences
EXAMPLE

Determine the $f(x) = 0$ solution of the following equation for $f(x) = 2x^3 - x^2 - 8x - 11$, where the convergence condition is $0.00001$, the maximum number of convergences is $40$, and initial values are $x_0 = -5$, $x_1 = 5$.

<table>
<thead>
<tr>
<th>Method of bisection $f(x) = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: $f(x)$. $x_0$. $x_1$</td>
</tr>
<tr>
<td>2: $x$. loop</td>
</tr>
</tbody>
</table>

(**Parameter input selection**)

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon = 0.0000001$ (Parameter input selection)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Max loop $(n&gt;0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 30$ ?</td>
</tr>
</tbody>
</table>

(**Convergence condition input**)

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x), (n&gt;0)$</td>
</tr>
<tr>
<td>$n = 30$ ?</td>
</tr>
</tbody>
</table>

(**Convergence condition input**)

<table>
<thead>
<tr>
<th>Define function $f(x)$ ?</th>
</tr>
</thead>
</table>

(**Function/initial value input selection**)

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2x^3 - x^2 - 8x - 11$ (Function input)</td>
</tr>
</tbody>
</table>

| $f(x) = 2x^3 - x^2 - 8x - 11$ (Function input) |
| $x_0 = 0$ ? |

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x) = 2x^3 - x^2 - 8x - 11$ (Initial value $x_0$ input)</td>
</tr>
<tr>
<td>$x_1 = 5$ ?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x) = 2x^3 - x^2 - 8x - 11$ (Initial value $x_1$ input)</td>
</tr>
<tr>
<td>$x = \ldots$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x) = 2x^3 - x^2 - 8x - 11$ (Solution display)</td>
</tr>
<tr>
<td>$x = 2.7171$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method of bisection $f(x) = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: $f(x)$. $x_0$. $x_1$</td>
</tr>
<tr>
<td>2: $x$. loop</td>
</tr>
</tbody>
</table>

(**Return to initial display**)

This display indicates that the solution for the example equation is $2.7171$.

The message "not found" is displayed when a solution cannot be found.

$\text{f}(x) = x^2 + 1$

not found
Matrix operations make it possible to perform addition, subtraction, multiplication, scalar product, determinant, inverse matrix, and transposed matrix calculations.

**OPERATION**

| 5100 | Matrix A(2,2):B(2,2) | >A.B.D.I.T.K.+.-V.M.C.P?

The following process can be selected from the menu display illustrated above.

A: Definition of MATRIX A and data input
B: Definition of MATRIX B and data input
D: Determinant of MATRIX A (det A)
I: Inverse matrix of MATRIX A and assignment of result to MATRIX A (A⁻¹→A)
T: Transposed matrix of MATRIX A and assignment of result to MATRIX A (Aᵀ→A)
K: Scalar product of MATRIX A and assignment of result to MATRIX A (kA→A)
+: Addition of MATRIX A and MATRIX B and assignment of result to MATRIX A (A+B→A)
−: Subtraction of MATRIX A and MATRIX B and assignment of result to MATRIX A (A−B→A)
*: Multiplication of MATRIX A and MATRIX B and assignment of result to MATRIX A (A•B→A)
M: Assignment of MATRIX A contents to MEMORY MATRIX M (A→M)
L: Assignment of MEMORY MATRIX M contents to MATRIX A (M→A)
C: Exchange of MATRIX A and MATRIX B contents (A→B)
P: Display of MATRIX A contents
-: Help display

**MATRIX SET UP**

Select either **A** (MATRIX A) or **B** (MATRIX B) from the menu display for matrix set up.

**EXAMPLE 1**

Set up the 3-row by 4-column matrix shown to the right.

\[
\begin{pmatrix}
1 & 0 & 3 & 4 \\
2 & 1 & 0 & -1 \\
3 & 1 & -2 & 3
\end{pmatrix}
\]

| Matrix A(2,2):B(2,2) | >A.B.D.I.T.K.+.-V.M.C.P?

- **A**
- **3**
- **4**

**(MATRIX A specification)**

**(Row input)**

**(Column input)**

- A 2-row by 2-column \[
\begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}
\] matrix is automatically set up when this library is activated.
Now enter the elements in the sequence shown in the illustration to the right (1–12).

\[
\begin{bmatrix}
1 & 3 & 4 \\
2 & 1 & 0 & -1 \\
3 & 1 & -2 & -3 \\
\end{bmatrix}
\]

\[a(1.1) = 0\]
\[a(1.2) = 0\] (Matrix element entries)
\[a(1.3) = 0\]

The unit returns to the menu display once input of all of the elements is complete. At this point, it is advisable to review the values to confirm that input was performed correctly.

A

\[A(m.n) = A(3.4)\] (MATRIX A specification)
\[m ?\]
\[n ?\] (Press EX after confirmation)
\[a(1.1) = 1\]
\[a(1.2) = 0\]

\[a(3.3) = -2\]
\[a(3.4) = -3\] (Return to initial display)

CORRECTION

Errors discovered before the EX key is pressed can be corrected by simply entering the correct value and then pressing EX. After EX is pressed, press EX to return to the previous value display and then make necessary corrections.

* The P command can also be used to view matrix contents.

Matrix addition/subtraction/multiplication

**EXAMPLE 2**

\[A = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}\]

Perform A+B, A−B, A·B, and B·A for the two following matrices.

Perform the following operation from the menu display.

The results of most matrix operations are stored in MATRIX A, deleting any contents currently stored in MATRIX A. Therefore, it is advisable to first transfer the contents from MATRIX A to MATRIX M so they can be recalled if later required before performing a matrix operation.

Once matrix set up is complete, proceed with the following calculations.

- **A+B**

\[
\begin{align*}
A+B &= A(2,2) \\
&> A.B.D.I.T.K.+.-*M.L.C.P.
\end{align*}
\]

\[
\begin{align*}
a(1,1) &= 3 \\
\text{(Result display)}
\end{align*}
\]

\[
\begin{align*}
a(1,2) &= 4 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]

\[
\begin{align*}
a(2,1) &= 4 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]

\[
\begin{align*}
a(2,2) &= 2 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]

\[
\begin{align*}
\text{Matrix } A'(2,2) &= B(2,2) \\
&> A.B.D.I.T.K.+.-*M.L.C.P.
\end{align*}
\]

Here, the result of A+B is \[
\begin{bmatrix}
3 & 4 \\
4 & 2
\end{bmatrix}
\]

- **A-B**

\[
\begin{align*}
\text{Load } A &= M(2,2) \\
&> A.B.D.I.T.K.+.-*M.L.C.P.
\end{align*}
\]

\[
\begin{align*}
\text{Matrix } A'(2,2) &= B(2,2) \\
&> A.B.D.I.T.K.+.-*M.L.C.P.
\end{align*}
\]

\[
\begin{align*}
A-B &= A(2,2) \\
&> A.B.D.I.T.K.+.-*M.L.C.P.
\end{align*}
\]

\[
\begin{align*}
a(1,1) &= -1 \\
\text{(Result display)}
\end{align*}
\]

\[
\begin{align*}
a(1,2) &= -2 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]

\[
\begin{align*}
a(2,1) &= 0 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]

\[
\begin{align*}
a(2,2) &= 0 \\
\text{(Press } \text{Enter} \text{ after confirmation)}
\end{align*}
\]
Here, the result of $A - B$ is  
\[
\begin{pmatrix}
-1 & -2 \\
0 & 0
\end{pmatrix}
\]

* $A \cdot B$

Here, the result of $A \cdot B$ is  
\[
\begin{pmatrix}
4 & 4 \\
6 & 7
\end{pmatrix}
\]

* $B \cdot A$

Here, the result of $B \cdot A$ is  
\[
\begin{pmatrix}
8 & 5 \\
4 & 3
\end{pmatrix}
\]
EXAMPLE 3

Calculate the determinant for the following matrix.

\[
\begin{bmatrix}
2 & 1 \\
0 & -1 \\
1 & 3
\end{bmatrix}
\begin{bmatrix}
3 & -1 & 1 \\
0 & 2 & 1
\end{bmatrix}
+ \begin{bmatrix}
1 & 0 & 1 \\
2 & -3 & 0 \\
0 & 0 & 2
\end{bmatrix}
\]

First perform the multiplication in the first term by setting up the following matrices and then executing A\(\times\)B.

\[
A = \begin{bmatrix}
2 & 1 \\
0 & -1 \\
1 & 3
\end{bmatrix}, \quad B = \begin{bmatrix}
3 & -1 & 1 \\
0 & 2 & 1
\end{bmatrix}
\]

Next perform calculation for second term.
Here, the result of the calculation is
\[
\begin{bmatrix}
2 & 1 \\
0 & -1 \\
1 & 3 \\
\end{bmatrix}
\begin{bmatrix}
3 & -1 & 1 \\
0 & 2 & 1 \\
0 & 0 & 2 \\
\end{bmatrix}
+ 
\begin{bmatrix}
1 & 0 & 1 \\
2 & -3 & 0 \\
0 & 0 & 2 \\
\end{bmatrix}
= 
\begin{bmatrix}
7 & 0 & 4 \\
2 & -5 & -1 \\
3 & 5 & 6 \\
\end{bmatrix}
\]

Determinant, inverse matrix, and transposed matrix

**EXAMPLE 4**
Determine the determinant, inverse matrix and transposed matrix for the following 3-column/3-row matrix.
\[
\begin{bmatrix}
2 & 0 & 0 \\
3 & 1 & 2 \\
4 & 2 & 3 \\
\end{bmatrix}
\]

**D**

**Determinant (det A)**

\[
\text{Determinant A} = \ldots \\
\]

(Determinant)

\[
\text{Determinant A} = -2 \\
\]

(Result display)

(Return to menu display)

Here, the determinant of MATRIX A is -2.

**Inverse matrix (A^-1)**

\[
\text{Inverse A} = A \\
\]

(Inverse matrix)

\[
\text{Inverse A} = 0.5 \\
\]

(Inverse matrix display)
Here, the inverse matrix of MATRIX A \( (A^{-1}) \) is

\[
\begin{bmatrix}
0.5 & 0.5 \\
0 & -1 \\
0 & 2 \\
-1 & -1
\end{bmatrix}
\]

- Transposed matrix \( (A^t) \)

Here, the transposed matrix for MATRIX A \( (A^t) \) is

\[
\begin{bmatrix}
2 & 3 & 4 \\
0 & 1 & 2 \\
0 & 2 & 3
\end{bmatrix}
\]
Scalar product

EXAMPLE 5
Calculate the scalar products for the following matrices.

\[
\begin{bmatrix}
1 & 2 \\
2 & 1 \\
\end{bmatrix}
\begin{bmatrix}
3 & 1 \\
0 & 2 \\
\end{bmatrix}
\]

\[A = \begin{bmatrix}
1 & 2 \\
2 & 1 \\
\end{bmatrix}, \quad B = \begin{bmatrix}
3 & 1 \\
0 & 2 \\
\end{bmatrix}\]

Multiply MATRIX B by the result of MATRIX A times three.

\[
\text{Matrix A}(2,2) \cdot B(2,2)
\]

\[
\begin{bmatrix}
A & B & D & I & T & K & + & - & \cdot & . & M & L & C & P & ?
\end{bmatrix}
\]

\[
\text{MATRX A set up}
\]

\[
(\text{MATRX A set up})
\]

\[
(\text{MATRX A set up})
\]

\[
\text{MATRX A}(2,2) \cdot B(2,2)
\]

\[
\begin{bmatrix}
A & B & D & I & T & K & + & - & \cdot & . & M & L & C & P & ?
\end{bmatrix}
\]

\[
(\text{MATRX A set up})
\]

\[
(\text{MATRX A set up})
\]

Here, the result of the example calculation is

\[
\begin{bmatrix}
1 & 2 \\
2 & 1 \\
\end{bmatrix}
\begin{bmatrix}
3 & 1 \\
0 & 2 \\
\end{bmatrix} = \begin{bmatrix}
9 & 15 \\
18 & 12 \\
\end{bmatrix}
\]

HELP menu
Pressing \(\Box\) in the menu display produces a HELP display which explains the meaning of each command.

\[A : \text{Input A(m,n)}\]

\[B : \text{Input B(m,n)}\]

Pressing \(\Box, \Box\) or \(\Box\) at this time scrolls through the commands. Pressing either \(\Box\) or \(\Box\) returns to the initial display.
Matrix display

After performing matrix addition, subtraction, multiplication, scalar product, determinant, inverse matrix, and transposed matrix calculations, the result of the calculation (contents of MATRIX A) is shown on the display. As with the HELP menu, [1] and [2] ( [3] ) can be used to scroll through MATRIX A.

* The operation of [1] and [2] is identical, with display being performed in the same sequence as the matrix element input. The [3] key displays the elements in reverse sequence.
* The [6] key can be used from the menu display to display the contents of MATRIX A. [7], [8], [9], [0] and [1] can also be used as desired.

### EXAMPLE

<table>
<thead>
<tr>
<th>Matrix</th>
<th>A(2,2) : B(2,2)</th>
</tr>
</thead>
</table>
Determines the integral value of interval [a, b] of the function \( y = f(x) \) graphed below using Romberg's Method.

The following parameters are specified in order to determine the numeric integration using Romberg's Method.

- **a, b**: Interval
- **\( \epsilon \)**: Error conditions to determine number of divisions (\( \epsilon > | \text{area}_{n+1} - \text{area}_n | \) )
- **loop**: Maximum number of divisions (positive integer)

The initial value of the area is determined using the trapezoidal formula.

- The variable name for the function \( f(x) \) must be represented by \( x \).
- The value input for \( \epsilon \) must be \( 1E-90 \) or more. Since internal calculations are performed in 12 digits, smaller values have little meaning.

**OPERATION**

5200 \( \text{[LB]} \)

\[
\begin{array}{c}
\text{Romberg's method} / f(x)dx \ (a, b) \\
1 : f(x) \ (a, b) \\
2 : \epsilon, \ loop
\end{array}
\]

The menu display illustrated above appears when the library is activated. Either 1 or 2 should be selected in accordance with the type of processing to be performed.

1: Function \( f(x) \) specification/interval input
2: Error condition/maximum number of divisions input

**EXAMPLE**

Determine the integral values in intervals \([3, 5]\) when \( f(x) = \ln x \). The error condition (\( \epsilon \)) = 0.0001, and the maximum number of divisions (loop) is \( 2^{10} \).

<table>
<thead>
<tr>
<th>2</th>
<th>0.0001</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{Romberg's method} / f(x)dx \ (a, b) )</td>
<td>( \text{Define function} \ f(x) )</td>
<td>( n = 8 )</td>
</tr>
<tr>
<td>( \text{Err} \</td>
<td>\text{An+1}-\text{An}</td>
<td>&lt; \epsilon \ (\epsilon &gt; 0) )</td>
</tr>
<tr>
<td>( \epsilon = 0.00000001 )</td>
<td>(Error condition input)</td>
<td>(Function specification, interval input selection)</td>
</tr>
</tbody>
</table>
\[ \int_a^b \ln x \, dx \quad \text{(Function specification)} \]
\[ \int_a^b \ln x \, dx = \ln a - \ln b \quad \text{(Integral value display)} \]
\[ \int_a^b f(x) \, dx = 2.7514 \]

This display indicates that the integral value for the example is 2.7514.

The message "not found" is displayed when an integral value cannot be found.

\[ \int_a^b \ln x \, dx \quad \text{not found} \]

**IMPORTANT**

Depending on the type of integration function or the integration range, large errors may be generated in values obtained through integration. The following points should be carefully noted to ensure accurate integral values.

1. **Periodic Functions and Symmetric Functions**
   Perform calculations for each period or symmetrical cycle.

   \[
   \int_0^b f(x) \, dx = \int_0^{\frac{b}{4}} f(x) \, dx + \int_{\frac{b}{4}}^{\frac{b}{2}} f(x) \, dx + \int_{\frac{b}{2}}^{\frac{3b}{4}} f(x) \, dx + \int_{\frac{3b}{4}}^b f(x) \, dx
   \]
   In the graph to the left, \( n = 4 \).

2. **Positive/Negative Integral Values According to Integral**
   Divide into positive portion integral and negative portion, and calculate individually.

3. **Large Fluctuation in Integral Values Due to Minute Fluctuation in Integration Range**
   Divide the integral interval (make the interval smaller where the fluctuation is large), and calculate individually.
The differential equation expressed as \( \frac{dy}{dx} = f(x, y) \) returns \( x = a, y = y(a) \) as the initial condition to obtain the numeric solution.

In the figure to the left, initial condition \( x = a, y = y(a) \) is returned when the differential equation \( \frac{dy}{dx} \) for the unknown function \( y = f(x) \) is known, and the numeric solution for \( x \) in the unknown function is calculated.

**OPERATION**

**5220 [LIB]**

Define function \( \frac{dy}{dx} \)

**EXAMPLE**

Express the differential equation \( f(x, y) = \frac{3y}{1+x} \), (initial condition: \( y(0) = 1 \)) with a numeric solution where the step size is 0.1.

\[
3 \times \frac{y}{(1 + x)} \quad \text{EX} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \\
x_0 = 0 \quad \text{?} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \\
y_0 = 0 \quad \text{?} \\
\text{Step-size} \quad \Delta h \quad \Delta h > 0 \\
\text{Step-size} = 1 \quad ? \\
y = 1 \quad \text{EX} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \quad \text{EX} \\
y(0.1) = 1.3305833302 \\
\text{EX} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \quad \text{EX} \\
y(5) = 215.8911132 \\
\text{EX} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \quad \text{?} \\
\frac{dy}{dx} = \frac{3y}{1 + x} \quad \text{?}
\]

While the numeric solution is displayed, \( \text{[EX]} \) (or \( \text{EX} \)) displays the next numeric solution while \( \text{[EX]} \) displays the last numeric solution. The \( \text{[EX]} \) and \( \text{[EX]} \) keys return to the initial display. Also, the numeric solution can be displayed up to step size \( x \times n \) \( (1 \leq n \leq 50) \).
5230 LAGRANGE'S INTERPOLATION

An nth degree polynomial is created to connect \( n + 1 \) points on a plane, and the data are interpolated according to the polynomial. This unit is capable of handling points within the range of \( 2 \leq n \leq 200 \) (\( n \) = integer).

Determine the \( n \) polynomial for the curve which passes through the four points noted on the left when \( n = 4 \).

OPERATION

5230 UB

Lagrange's interpolation
1: set data

The following two operations can be selected from the initial display:
1 : Data interpolation
2 : Input of \( n \) number of points

EXAMPLE

Create a 3rd degree polynomial which connects the following three points and determine the value when \( x = 4 \).

\( P_1 \ (1, 3), P_2 \ (3, 1), P_3 \ (5, 2) \)

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Number of data ( n = 2 )</td>
</tr>
<tr>
<td>3</td>
<td>Number of data = 3 ( x_1 = 0 )</td>
</tr>
<tr>
<td>1</td>
<td>Number of data = 3 ( y_1 = 0 )</td>
</tr>
<tr>
<td>3</td>
<td>Number of data = 3 ( x_2 = 0 )</td>
</tr>
<tr>
<td>3 EEX 1 EEX 5 EEX 2 EEX</td>
<td>Lagrange's interpolation (Coordinates of remaining points)</td>
</tr>
<tr>
<td>1</td>
<td>Lagrange's interpolation (Data interpolation)</td>
</tr>
<tr>
<td>4 EEX</td>
<td>Lagrange's interpolation ( x = 1.125 ) (x-data input and 1.125 is obtained for ( y ))</td>
</tr>
<tr>
<td>EEX</td>
<td>Lagrange's interpolation ( x ) (x-data input requested)</td>
</tr>
<tr>
<td>EEX</td>
<td>Lagrange's interpolation 1: set data (Return to menu display)</td>
</tr>
</tbody>
</table>
Here it can be seen that a value of 1.125 is obtained when \( x = 4 \).
* The "not found" message as illustrated below appears when interpolation is not performed using the nth degree polynomial.

![Graph showing the gamma function \( \Gamma(x) \)]

**5250 GAMMA FUNCTION \( \Gamma(x) \)**

Determines the value of the gamma function within the range of \( 0 < x \leq 70 \) with six significant digits.

The gamma function is expressed as the graph shown on the left.

**OPERATION**

5250 [LIB]

\[
\text{ Gamma function } \quad (0 < x \leq 70)
\]

\[ x = 1 ? \]

**EXAMPLE**

Determine the value of the gamma function when \( x = 3 \).

\[
\begin{array}{c}
3 \text{ EXE} \\
\Gamma(3) = 2 \\
\text{ EXE}
\end{array}
\]

(x value input) (Result display) (Return to initial display)

Here, the value of the gamma function is 2.
* A total of six entries (including decimal points) can be made for input of \( x \).
Determine the elementary solution $J_n(x)$ of the Bessel differential equation $\frac{d^2y}{dx^2} + \frac{1}{x} \cdot \frac{dy}{dx} + (1 - \frac{n^2}{x^2}) \cdot y = 0$ within the range of $0 \leq n \leq 9$ (integer), $0 \leq x \leq 30$ (condition of $x$) with six significant digits.

![Graph of Bessel functions $J_0(x)$ and $J_1(x)$]

**OPERATION**

<table>
<thead>
<tr>
<th>5260 UB</th>
<th>$J_n(x)$</th>
<th>$0 \leq n \leq 9 : 0 \leq x \leq 30$</th>
<th>$n ?$</th>
<th>$x ?$</th>
<th>$J = \ldots$</th>
</tr>
</thead>
</table>

**EXAMPLE**

Determine the Bessel function $J_n(x)$ when $n = 2$ and $x = 3$.

| 2 EX 3 EX | $J_n(x)$ | $0 \leq n \leq 9 : 0 \leq x \leq 30$ | $n ? 2$ | $x ? 3$ | $J = \ldots$ | $J = 0.486091$ | $J = \ldots$ | Return to initial display |

Here, the Bessel function value is 0.486091.

* A total of six entries (including decimal points) can be made for input of $x$. 

---

209
BESSEL FUNCTION Yn(x)

Determines the elementary solution Yn(x) of the Bessel differential equation \( \frac{d^2y}{dx^2} + \frac{1}{x} \cdot \left(1 - \frac{n^2}{x^2}\right)y = 0 \) within the range of 0 ≤ n ≤ 9 (integer), 0 < x ≤ 30 (condition of x) with six significant digits.

\[ y = Y_0(x) \quad y = Y_1(x) \]

\[ y = Y_0(x) \quad n = 0 \quad y = Y_1(x) \quad n = 1 \]

OPERATION

5270 UB

\[ Y_n(x) \quad (0 \leq n \leq 9, \ 0 < x \leq 30) \]

n? - : x ?

EXAMPLE

Determine the Bessel function Yn(x) when n = 3 and x = 4.

\[ n = 3 \quad x = 4 \]

\[ Y_n(x) \quad (0 \leq n \leq 9, \ 0 < x \leq 30) \]

\[ n = 3 \quad x = 4 \]

Y = -0.182022

Here, the Bessel function value is −0.182022.

* A total of six entries (including decimal points) can be made for input of x.
MODIFIED BESSEL FUNCTION $I_n(x)$

Determines the elementary solution $I_n(x)$ of the modified Bessel differential equation $\frac{d^2y}{dx^2} + \frac{1}{x} \cdot \frac{dy}{dx} - \left(1 + \frac{n^2}{x^2}\right) y = 0$ within the range of $0 \leq n \leq 9$ (integer), $0 \leq x \leq 10$ (condition of $x$) with six significant digits.

\[
y = I_0(x) : n = 0
y = I_1(x) : n = 1
\]

OPERATION

$5280$ [LB]

<table>
<thead>
<tr>
<th>$I_n(x)$</th>
<th>$0 \leq n \leq 9 \quad 0 \leq x \leq 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n ?$ : $x ?$</td>
<td>$i =$</td>
</tr>
</tbody>
</table>

EXAMPLE

Determine the modified Bessel function when $n = 3$ and $x = 5$.

3 [EXE] 5 [EXE]

<table>
<thead>
<tr>
<th>$I_n(x)$</th>
<th>$0 \leq n \leq 9 \quad 0 \leq x \leq 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n ? 3$ : $x ? 5$</td>
<td>$i =$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$I_n(x)$</th>
<th>$0 \leq n \leq 9 \quad 0 \leq x \leq 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n ? 3$ : $x ? 5$</td>
<td>$i =$</td>
</tr>
</tbody>
</table>

Here, the modified Bessel function value is 10.3312.

A total of six entries (including decimal points) can be made for input of $x$. 
MODIFIED BESSEL FUNCTION $K_n(x)$

Determines the elementary solution $K_n(x)$ of the modified Bessel differential equation $\frac{d^2y}{dx^2} + \frac{1}{x} \frac{dy}{dx} - \left(1 + \frac{n^2}{x^2}\right)y = 0$ within the range of $0 \leq n \leq 9$ (integer), $0 < x \leq 10$ (condition of $x$) with six significant digits.

\[ y = K_0(x) \quad y = K_1(x) \quad y = K_n(x) : n = 0 \quad y = K_n(x) : n = 1 \]

**OPERATION**

5290 LIB

**EXAMPLE**

Determine the modified Bessel function $K_n(x)$ when $n = 4$ and $x = 6$.

\[
\begin{array}{c|c|c}
\text{Kn} (x) & (0 \leq n \leq 9; 0 < x \leq 10) & \text{K} = \\
n \neq 0 & x \neq 0 & \\
\end{array}
\]

\[
\begin{array}{c|c|c|c|c|c|c|c}
4 & 6 & 4 & 6 & = 4 & 6 & 0 & 0416387 & \text{K} = \\
\end{array}
\]

Here, the modified Bessel function value is 0.00416387.

* A total of six entries (including decimal points) can be made for input of $x$. 
Complex number calculations encompass arithmetic operations, and to determine absolute values, arguments, squares, square roots, and reciprocal numbers. This unit is capable of a wide variety of complex number calculations, with the allowable range of input value < 1E50.

**OPERATION**


The complex number menu display allows selection of the following processes:

A: Input of complex number A (a + bi)
G: Absolute value (r) and arguments (θ) for complex number A (resulting angle unit determined by current mode setting)
I: Reciprocal number for complex number A \(1/(a + bi) \rightarrow (a + bi)\)
S: Square root of complex number A \(\sqrt{a + bi} \rightarrow (a + bi)\)
^: Square of complex number A \((a + bi)^2 \rightarrow (a + bi)\)
+: Addition of complex number A and complex number B \((a + bi) + (c + di) \rightarrow (a + bi)\)
 -: Subtraction of complex number A and complex number B \((a + bi) - (c + di) \rightarrow (a + bi)\)
*: Multiplication of complex number A and complex number B \((a + bi) \times (c + di) \rightarrow (a + bi)\)
/: Division of complex number A and complex number B \((a + bi) \div (c + di) \rightarrow (a + bi)\)
M: Assigns contents of complex number A to complex number memory M (e + fi) \((a + bi) \rightarrow (e + fi)\)
L: Assigns contents of complex number memory M (e + fi) to complex number A \((a + bi) \leftarrow (e + fi)\)
C: Exchanges contents of complex number A and complex number B \((a + bi) \leftrightarrow (c + di)\)
.: Help (explanation of each operation)

**Complex Number Specification**

Complex number specification is performed by pressing [A] while in the menu display.

**EXAMPLE**

Assign 5 + 7i to complex number A.

```
A
```

```
5 [EU]
```

```
7 [EU]
```

```
```

```
Complex number A (a + bi)
```

```
e = 0 ?
```

```
5 + 7i
```

```
Complex number A (a + bi)
```

```
b = 0 ?
```

```
```

```
Complex number A (a + bi)
```

```
(e + fi)
```

```
Complex number A (a + bi)
```

```
(e + fi)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
Complex number A (a + bi)
```

```
(c + di)
```

```
• Arithmetic Operations

**EXAMPLE**

Perform the following operations:

$$(2 + 3i) + (3 - 2i)$$

- **A**
  - Complex number $A(a+bi)$
  - $a = 0$ ?

- **2 EX 3 EX**
  - $2 + 3i$
  - $<A.G.I.S.+.-.i./M.L.C.?$

- **+**
  - Complex number $B(c+di)$
  - $c = 0$ ?

- **3 EX 2 EX**
  - $5 + i$
  - $>A.G.I.S.+.-.i./M.L.C.?$

This display indicates $(2 + 3i) + (3 - 2i) = 5 + i$.
The same procedure can be performed for subtraction, multiplication and division.

• Absolute Values/Arguments

**EXAMPLE**

Determine the absolute value ($r$) and argument ($\theta$) for $(1 + 2i)$.
Angle unit: DEG (AE 4)

![Imaginary number axis](image)

- **A**
  - Complex number $A(a+bi)$
  - $a = 0$ ?

- **1 EX 2 EX**
  - $1 + 2i$
  - $>A.G.I.S.+.-.i./M.L.C.?$

- **G**
  - $r = 2.236067977$
  - $\theta = 63.43494882$

Here, the absolute value ($r$) for $(1 + 2i)$ is $2.236067977$, and the argument is $63.43494882$ (DEG). The resulting angle unit is determined by the current ANGLE mode setting.

* The angle unit is specified as follows:

- **DEG** : Degrees
- **GRD** : Radians
- **GRS** : Grads
- Square/Square Root/Reciprocal number

**EXAMPLE**

Calculate the following:
1. \((2 + i)^2\)
2. \(\sqrt{-7 + 24i}\)
3. \(\frac{1}{3 + 2i}\)

1. **Square**
   
   A
   
   \[ \text{Complex number } A(a + bi) \]
   
   \[ a = 0 \text{ ?}_- \]
   
   \[ \frac{2}{-A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Specification of complex number input)
   
   \[ \frac{2}{-A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Input of complex number A)

   This display indicates \((2 + i)^2 = 3 + 4i\).

2. **Square Root**
   
   A
   
   \[ \text{Complex number } A(a + bi) \]
   
   \[ a = 0 \text{ ?}_- \]
   
   \[ -7 + 24i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Specification of complex number input)
   
   \[ -7 + 24i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Input of complex number A)
   
   S
   
   \[ 3 + 4i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Square root)

   This display indicates \(\sqrt{-7 + 24i} = 3 + 4i\).

3. **Reciprocal Number**
   
   A
   
   \[ \text{Complex number } A(a + bi) \]
   
   \[ a = 0 \text{ ?}_- \]
   
   \[ 3 + 2i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Specification of complex number input)
   
   \[ 3 + 2i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Input of complex number A)
   
   I
   
   \[ 0.2307692 \]
   
   \[ -.0 .1538462i \]
   
   \[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]
   
   (Reciprocal number)

- Memory Calculations

**EXAMPLE**

Perform the following calculations using the memory function:

\[(3 + 2i) + (4 + 6i)\]
\[(3 + 2i) - (-3 + 9i)\]

A

\[ \text{Complex number } A(a + bi) \]

\[ a = 0 \text{ ?}_- \]

\[ 3 + 2i \]

\[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]

(M)

\[ 3 + 2i \]

\[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]

(Assigns complex number A to complex number memory)

+\n
\[ \text{Complex number } B(c + di) \]

\[ c = 0 \text{ ?}_- \]

\[ 7 + 8i \]

\[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]

(Addition)

4 EX 6 EX

\[ 7 + 8i \]

\[ >A.G.I.S.^+.+.+..} . M.L.C ?_- \]

(Assigns 4 + 6i to B)

This display indicates \((3 + 2i) + (4 + 6i) = 7 + 8i\).
This display indicates $(3 + 2i) - (-3 + 9i) = 6 - 7i$.

**Exchange**

**Example**

Set the following two complex numbers for complex numbers $A$ and $B$:

$(5 + 2i), (3 + 4i)$

The above operation sets $5 + 2i$ for complex number $A$, and $3 + 4i$ for complex number $B$.

* Help Display

Pressing $\rightarrow$ while in the menu display produces an explanation of each command.

At this time $\leftarrow$, $\rightarrow$, and $\uparrow$ can be used to scroll the display. Each press of $\rightarrow$ (or $\leftarrow$) advances to the next command, while pressing $\uparrow$ returns to the previous command. Pressing $\downarrow$ or $\rightarrow$ returns to the menu display. The menu display is also returned to after the final command is displayed.
Binary, decimal and hexadecimal calculations encompass basic arithmetic operations, logical operations, twos complement, logical shift, and conversions.

This unit is capable of combining binary, decimal and hexadecimal values, with the allowable range values being $-2147483648 \sim 2147483647$ (32-bit).

**OPERATION**

The binary/decimal/hexadecimal calculation menu display allows selection of the following processes:

1. Value input
2. Converts displayed value to binary number
3. Converts displayed value to decimal number
4. Converts displayed value to hexadecimal number
5. Addition
6. Subtraction
7. Multiplication
8. Division
9. AND (logical product)
10. OR (logical sum)
11. XOR (exclusive logical sum)
12. NOT (negation)
13. Twos complement
14. Logical shift left
15. Logical shift right
16. Help (explanation of each operation)

**Operations and Display**

1. The following indicators in the upper left of the display in the menu indicate the current base mode setting:
   - [DEC] : Decimal mode
   - [HEX] : Hexadecimal mode
   - Blank : Binary mode

2. Entering values besides 0 and 1 for binary calculations, values besides 0 - 9 for decimal calculations, values besides 0 - A - F (upper case or lower case) for hexadecimal calculations, or values greater than 32 bits causes the entered value to be disregarded.

   Binary, decimal and hexadecimal values may be used in combination in a single calculation.
EXAMPLE

The following operations may be used to enter values regardless of the current base mode setting:

15, D : Decimal 15 (hexadecimal F, binary 1111)
15, H : Hexadecimal 15 (decimal 21, binary 10101)
1010, B : Binary 1010 (decimal 10, hexadecimal A)

Results are always displayed using the current base mode setting.

Arithmetic Operations

EXAMPLE

Perform the following calculations:

1. 1011001b + 1100b
2. 2ACH × 1Bh
3. FF00h + 1010b

1)

This display indicates 1011001b + 1100b = 1100101b

2)

This display indicates 2ACH × 1Bh = 4824h.

The same procedure can be performed for subtraction and division.

3)


This display indicates FF00H + 1010B = FF0AH

- Logical Operations

**EXAMPLE**

Perform the following operations for \(A = 110101B\) and \(B = 101111B\).

1. \(A \text{ OR } B\) (logical sum)  
2. \(A \text{ AND } B\) (logical product)  
3. \(A \text{ XOR } B\) (exclusive logical sum)  
4. \(A \text{ NOT}\) (negation)

\[ \begin{align*}
\text{(HEX)} & \quad \text{C000FF00} + - \\
\text{(Binary mode)} & \quad \text{1B.D.H.} + - x / A.O.X.N.C.L.R?\
\text{(Value input specification)} & \quad \text{Input data x (B.D.H) [BIN]}
\end{align*} \]

This display indicates \(A \text{ OR } B = 111111B\).

\[ \begin{align*}
\text{(HEX)} & \quad \text{C000FF0A} \\
\text{(Value input)} & \quad \text{1B.D.H.} + - x / A.O.X.N.C.L.R?
\end{align*} \]

This display indicates \(A \text{ AND } B = 100101B\).

\[ \begin{align*}
\text{(HEX)} & \quad \text{C000FF0A} \\
\text{(Value input specification)} & \quad \text{1B.D.H.} + - x / A.O.X.N.C.L.R?
\end{align*} \]

This display indicates \(A \text{ XOR } B = 11010B\).

\[ \begin{align*}
\text{(HEX)} & \quad \text{C000FF0A} \\
\text{(Value input specification)} & \quad \text{1B.D.H.} + - x / A.O.X.N.C.L.R?
\end{align*} \]
This display indicates NOT A = 111111111111111111111111010101010a.

**Complement/Shift Operations**

**EXAMPLE**

Perform the following operations:

1. Twos complement of 11001010s
2. 1-bit logical shift left of 110000a
3. 2-bit logical shift right of 1FCa

1.

```
[DEC] 0
>1.B.D.H.+.- x/.A.O.X.N.C.L.R7-
```

(Binary mode)

This display indicates that the twos complement of 11001010s is 111111111111111111111111010101010a.

2.

```
Input data x (.B.D.H) [BIN]
>1.B.D.H.+.- x/.A.O.X.N.C.L.R7-
```

(Shift left)

This display indicates that shifting 110000a one bit to the left results in 1100000a.

3.

```
[HEX] 00000060
>1.B.D.H.+.- x/.A.O.X.N.C.L.R7-
```

(2-bit shift right)

This display indicates that shifting 1FCa two bits to the right results in 7Fc.
**Base conversion**

**EXAMPLE**

Convert the hexadecimal value AF3C to its decimal and binary equivalents.

<table>
<thead>
<tr>
<th>(DEC)</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1.B.D.H.+.-.z./.A.O.X.N.C.L.R?</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(HEX)</th>
<th>0000AF3C</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1.B.D.H.+.-.z./.A.O.X.N.C.L.R?</td>
<td></td>
</tr>
</tbody>
</table>

**Value input specifications**

**Value input**

<table>
<thead>
<tr>
<th>(HEX)</th>
<th>0000AF3C</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1.B.D.H.+.-.z./.A.O.X.N.C.L.R?</td>
<td></td>
</tr>
</tbody>
</table>

**Decimal mode**

<table>
<thead>
<tr>
<th>(DEC)</th>
<th>44860</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1.B.D.H.+.-.z./.A.O.X.N.C.L.R?</td>
<td></td>
</tr>
</tbody>
</table>

**Binary mode**

<table>
<thead>
<tr>
<th>000000000000000010101110111100</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1.B.D.H.+.-.z./.A.O.X.N.C.L.R?</td>
</tr>
</tbody>
</table>

This display indicates that the decimal equivalent of hexadecimal AF3C is 44860, and the binary equivalent is 1010111100111100a.

**Help Display**

Pressing C while in the menu display produces an explanation of each command.

<table>
<thead>
<tr>
<th>1 : input data</th>
</tr>
</thead>
<tbody>
<tr>
<td>B : binary mode</td>
</tr>
</tbody>
</table>

At this time, EX and C can be used to scroll the display. Each press of EX (C) advances to the next command, while pressing X returns to the previous command. Pressing C or C returns to the menu display. The menu display is also returned to after the final command is displayed.
STRAIGHT LINE PASSING THROUGH TWO POINTS

Determines the straight line \( y = ax + b \) which passes through points \( P_1(x_1, y_1) \) and \( P_2(x_2, y_2) \) on a plane.

\[
y = ax + b
\]

**OPERATION**

\[
y = ax + b \quad \text{and} \quad (x_1 \cdot y_1), (x_2 \cdot y_2)
\]

**EXAMPLE**

Determine the line which passes through points \( P_1(2,5) \) and \( P_2(6,4) \)

\[
\begin{align*}
2 & \times \ 5 & \times & \times \\
& \times & \times & \times \\
& \times & \times & \times \\
& \times & \times & \times \\
& \times & \times & \times \\
& \times & \times & \times \\
\end{align*}
\]

\[
y = ax + b \quad \text{and} \quad (x_1 \cdot y_1), (x_2 \cdot y_2)
\]

\[
\begin{align*}
\text{P1 input} & : \quad y = 4x + b \quad \text{and} \quad (x_1 \cdot y_1), (x_2 \cdot y_2) \\
\text{a value displayed following P1 input} & : \quad b = 5 \ 5 \\
\text{b value displayed} & : \quad b = 5 \ 5 \\
\text{Redisplay of a value} & : \quad a = 0 \ 25 \\
\text{Redisplay of b value} & : \quad b = 5 \ 5 \\
\text{Return to initial display} & : \quad x_1 = 2 \ ?
\end{align*}
\]

Here, the straight line is \( y = -0.25x + 5.5 \).
Determines the angle of intersection created by the two lines $y_1 = ax + b$ and $y_2 = cx + d$. The calculated angle for $y_1$ and $y_2$ is within the range of $-90^\circ < \theta < 90^\circ$. The resulting angle unit is determined by the current angle mode setting.

* The angle unit is specified as follows:
  - [A]: Degrees
  - [B]: Radians
  - [C]: Grads

**Operation**

**Example**

Determine the angle of intersection (in DEG mode) for the straight lines $y_1 = \frac{1}{2}x + 2$ and $y_2 = 3x + 8$.

```
1 [E] 2 [E] 3
\text{Angle (} \theta \text{) } \rightarrow \text{ } y = ax + b \cdot y = cx + d
\theta = 0 \text{° ?}
```

**Example**

Determine the angle of intersection (in DEG mode) for the straight lines $y_1 = 4x + 5$ and $y_2 = 4x + 7$.

```
4 [E] 4
\text{Angle (} \theta \text{) } \rightarrow \text{ } y = ax + b \cdot y = cx + d
\theta = 0 \text{° ?}
```

223
**EXAMPLE**

Determine the angle of intersection (in DEG mode) for the straight lines \( y_1 = \frac{1}{2}x + 3 \) and 
\( y_2 = -2x + 4 \).

1 \( \square \) 2 \( \triangle \) - 2  
\[ \text{Angle} (\theta) \quad y = ax + b \cdot y = cx + d \]  
\[ c = 0 \quad ? - 2 \]  
(Entry of each line's slope)

\[ \text{Angle} (\theta) \quad y = ax + b \cdot y = cx + d \]  
\[ \text{Right angle} \]  
(Indicates angle of intersection is right angle)

\[ \text{Angle} (\theta) \quad y = ax + b \cdot y = cx + d \]  
\[ a = 0 \quad .5 \quad ? \]  
(Return to initial display)

---

**5530**  
**DISTANCE BETWEEN POINT AND STRAIGHT LINE**

Determines length D of a perpendicular line from point P \((x_1, y_1)\) and straight line \( y = ax + b \).

![Diagram](image)

**OPERATION**

5530 \( \square \)  
\[ \text{Distance} \quad y = ax + b \cdot (x_1, y_1) \]  
\[ a = 0 \quad ? \]  

**EXAMPLE**

Determine length D of a perpendicular line from point P \((6, 4)\) to straight line \( y = 5x + 2 \).

5 \( \square \) 2 \( \triangle \)  
\[ \text{Distance} \quad y = ax + b \cdot (x_1, y_1) \]  
\[ x_1 = 6 \quad ? \]  
(Straight line's slope and intercept)

6 \( \square \) 4 \( \triangle \)  
\[ \text{Distance} \quad y = ax + b \cdot (x_1, y_1) \]  
\[ b = 5.491251784 \]  
(Distance displayed when coordinates of P are input)

\[ \text{Distance} \quad y = ax + b \cdot (x_1, y_1) \]  
\[ a = 5 \quad ? \]  
(Return to initial display)

Here, the length of the perpendicular line is 5.491251784.
Determines coordinates of point \( P_2 \) \((X, Y)\) when a rotation of angle \( \theta \) occurs from point \( P_1 \) \((x_1, y_1)\). The angle unit is determined by the current angle mode setting.

The angle unit is specified as follows:

- **4** : Degrees
- **5** : Radians
- **6** : Grads

OPERATION

5540 [LB]

\[(X, Y) \rightarrow (x, y), \text{angle}(\theta)\]

\[x = 0 \quad ?\]

EXAMPLE

Determine the coordinates of point \( P_2 \) \((X, Y)\) for rotation \( \theta \) of 45 (in DEG mode) from point \( P_1 \) \((4, 8)\).

\[
\begin{align*}
4 & \text{ [EX]} 8 \text{ [EX]} \quad \text{(P\_1 coordinates)} \\
45 & \text{ [EX]} \\
(X, Y) & \rightarrow (x, y), \text{angle}(\theta) \quad \text{(X-coordinate displayed when angle input)} \\
X & = 2.628427125 \\
Y & = 8.485281374 \\
\text{(Redisplay of X-coordinate)} \\
\text{(Redisplay of Y-coordinate)} \\
\text{(Return to initial display)}
\end{align*}
\]

Here, the coordinates of \( P_2 \) are \((-2.828427125, 8.485281374)\).

The result is displayed in the sequence of \( X, Y, \) and the display can be scrolled to view following values using \([\text{EX}](\text{or } \text{ [EX]}))\), and previous values can be viewed using \([\text{EX}])\).
Determine the equation \( (x - a)^2 + (y - b)^2 = r^2 \) for a circle passing through the points \( P_1(x_1, y_1) \), \( P_2(x_2, y_2) \), \( P_3(x_3, y_3) \).

**Example**

Determine the equation \( (x - a)^2 + (y - b)^2 = r^2 \) for the circle which passes through points \( P_1(3, 6) \), \( P_2(5, 4) \), \( P_3(6, 2) \).

Here, the equation for the circle becomes \( (x + 1.5)^2 + (y + 0.5)^2 = 7.90569415^2 \). The result is displayed in the sequence of \( a, b, r \), and the display can be scrolled to view following values using \( \text{C} \) (or \( \text{EX} \)), and previous values using \( \text{DC} \).
LENGTH OF TANGENT LINES FROM A POINT TO A CIRCLE

Determines length l from point P \((x_1, y_1)\) to a circle expressed by the equation \((x-a)^2 + (y-b)^2 = r^2\).

```
OPERATION
5560 [LB]  \((x-a)^2 + (y-b)^2 = r^2 \cdot (x_1, y_1)\)
```

**EXAMPLE**

Determine the length l of a tangent line from point P \((2, 5)\) to a circle with center point O \((6, 2)\) and a radius of 4.

```
6 EX 2 EX 4 EX 5 EX 2 EX
\((x-a)^2 + (y-b)^2 = r^2 \cdot (x_1, y_1)\) (Coordinates of circle’s center point)
\(r = 4\) (Circle’s radius)
\((x-a)^2 + (y-b)^2 = r^2 \cdot (x_1, y_1)\) (Tangent line length display following input of point coordinates)
l = length = 3
\(a = 6\) (Return to initial display)
```

Here, the length of tangent line l is 3.

The message "not found" appears on the display when the coordinates of point P are within the circle.
Determine the equations for two lines $y_1 = cx + d$, $y_2 = ex + f$, and their points of tangency $P_2(x_2, y_2)$, $P_3(x_3, y_3)$ from point $P_1(x_1, y_1)$ to circle $O$ represented by the equation $(x-a)^2 + (y-b)^2 = r^2$.

**OPERATION**

5570

\[(x-a)^2 + (y-b)^2 = r^2 \cdot (x_1, y_1)\]

\[x = 0 \ ? \]

**EXAMPLE**

Determine the equations for tangent lines and points of tangency from point $P_1(1, 2)$ to a circle centered on point $O(4, 3)$ with a radius of 2.

\[(x_1, y_1)^2 + (y_1-b)^2 = r^2 \cdot (x_1, y_1)\]

\[x_1 = 0 \ ? \]

\[(x_2, y_2)^2 + (y_2-b)^2 = r^2 \cdot (x_2, y_2)\]

\[x_2 = 2.310102251\]

\[(x_3, y_3)^2 + (y_3-b)^2 = r^2 \cdot (x_3, y_3)\]

\[x_3 = 3.288997949\]

\[y_3 = 1.185961564\]

\[x_2 = 0.797589797\]

\[y_2 = 2.375958587\]

\[e = 0.375795857\]
Here, the two points of tangency are P2 (2.310102051, 4.069693846), P3 (3.289897949, 1.130306154). The equations for the lines which pass through these points are:

\[ y_2 = 1.579795897x + 0.4202041029 \]
\[ y_3 = -0.3797958971x + 2.379795897 \]

The result is displayed in the sequence of \( x_2 \), \( y_2 \), \( c \), \( d \), \( x_3 \), \( y_3 \), \( e \), \( f \), and the display can be scrolled to view following values using [X] (or [EX]) and previous values using [C].

Corresponding values of \( c \) and \( e \) are omitted when the equations for the tangent lines are parallel.

---

### 5600

**AREA OF A TRIANGLE**

Determines the area (S) of a triangle using one of the three following formulas:

1. \( S = \frac{1}{2}ah \)
2. \( S = \frac{ab \cdot \sin \theta}{2} \) (The result depends on the currently specified angle unit.)
3. \( S = \sqrt{s(s-a)(s-b)(s-c)} \) \( s = \frac{1}{2} (a+b+c) \)

**OPERATION**

5600 [LB]  

Area [triangle] 1: a \cdot h / 2 2: a \cdot b \cdot \sin \theta / 2 3: f (a, b, c)

**EXAMPLE**

Determine the area of triangle \( a = 10 \), \( h = 5 \).

1  

10 [EX] 5 [EX]  

a \cdot h / 2  

a: base h: height  

Area = 25 \( a \): base \( h \): height  

(Triangle’s base and height)  

Area [triangle] 1: a \cdot h / 2 2: a \cdot b \cdot \sin \theta / 2 3: f (a, b, c)  

(Return to menu display)

Here, the area of the triangle is 25.
EXAMPLE
Determine the area of triangle \((a = 10, \ b = 5, \ \theta = 30 \, (\text{DEG}))\).

\[
\begin{array}{c|c|c}
2 & \frac{a \cdot b \cdot \sin \theta}{2} & \text{a \cdot b \cdot \sin \theta} \\
3 & \text{Area} & 12.5 \\
\text{EX} & \text{Area (triangle)} & \text{2: \frac{a \cdot b \cdot \sin \theta}{2}} \\
\end{array}
\]

(Two sides and included angle)

Here, the area of the triangle is 12.5.

EXAMPLE
Determine the area of triangle \((a = 5, \ b = 4, \ c = 3)\).

\[
\begin{array}{c|c|c}
3 & \sqrt{s(s-a)(s-b)(s-c)} - s = \frac{(a+b+c)}{2} & \text{a \cdot b \cdot \sin \theta} \\
\text{EX} & \text{Area (triangle)} & 6 \\
\text{EX} & \text{Area (triangle)} & \text{2: \frac{a \cdot b \cdot \sin \theta}{2}} \\
\end{array}
\]

(Three sides)

Here, the area of the triangle is 6.

5605
AREA OF A TRAPEZOID

Determines the area \((S)\) of a trapezoid using the following formula:

\[
S = \frac{(a + b) \cdot h}{2}
\]

OPERATION

\[
\begin{array}{c|c|c}
5605 & \frac{(a + b) \cdot h}{2} & \text{a \cdot b \cdot \text{base} \cdot h \cdot \text{height}} \\
\text{EX} & \text{Area} & 30 \\
\text{EX} & \text{Area} & \text{2: \frac{(a + b) \cdot h}{2}} \\
\end{array}
\]

(Bases, height)

Here, the area of the trapezoid is 30.
Determines the area (S) of a parallelogram using one of the two following formulas:

\[ S = ah \]
\[ S = ab \cdot \sin \theta \]  
(The result depends on the currently specified angle mode.)

**OPERATION**

5610 [LB]

<table>
<thead>
<tr>
<th>Area (Parallelogram)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: ( a \cdot h )</td>
</tr>
<tr>
<td>2: ( a \cdot b \cdot \sin \theta )</td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the area of parallelogram (\( a = 10 \), \( h = 5 \)).

\[ \begin{align*}
| \text{Base and height} | \\
| a = 0 \ ?  \quad \text{a \ : base} \quad \text{h \ : height} | \\
| \text{Area} = 50 | \\
| \text{Return to initial display} |
\end{align*} \]

Here, the area of the parallelogram is 50.

**EXAMPLE**

Determine the area of parallelogram (\( a = 10 \), \( b = 6 \), \( \theta = 30 \) (DEG)).

\[ \begin{align*}
| \text{Two sides and included angle} | \\
| a \cdot b \cdot \sin \theta \ ? \quad a \ : \ side \quad b \ : \ side | \\
| \text{Area} = 30 | \\
| \text{Return to menu display} |
\end{align*} \]

Here, the area of the parallelogram is 30.
Determines the area ($S$) of a circle using the following formula:

$$S = \pi r^2$$

**OPERATION**

5615 [UB] \[
\begin{array}{c}
\times r^2 \\
\text{r: radius} \\
\text{r = 5} \\
\end{array}
\]

**EXAMPLE**

Determine the area of a circle with radius $r = 5$.

\[
\begin{array}{c}
\times r^2 \\
\text{Area = 78.53981634} \\
\text{r: radius} \\
\text{r = 5} \\
\end{array}
\]

(Circle's radius) (Return to initial display)

Here, the area of the circle is 78.53981634.
Determine the area (S) of a sector using one of the following formulas:

1. \( S = \frac{1}{2} \cdot r \cdot \theta \)
2. \( S = \pi r^2 \cdot \frac{\theta}{360} \)  
   (Angle unit = degrees)

**OPERATION**

**5620**

<table>
<thead>
<tr>
<th>Area (sector)</th>
<th>( \frac{1}{2} \cdot r \cdot \theta )</th>
</tr>
</thead>
</table>

**EXAMPLE**

Determine the area of sector \((l = 6, r = 8)\).

<table>
<thead>
<tr>
<th>( \frac{1}{2} \cdot l \cdot \text{circular arc} )</th>
<th>( r \cdot \text{radius} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l = 6 )</td>
<td>( r = 8 )</td>
</tr>
</tbody>
</table>

\( \text{Area} = 24 \)

**EXAMPLE**

Determine the area of sector \((r = 8, \theta = 30 \text{ (DEG)})\).

<table>
<thead>
<tr>
<th>( r \cdot \theta \cdot \frac{1}{360} )</th>
<th>( r \cdot \text{radius} )</th>
<th>( \theta \cdot \text{DEG} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r = 8 )</td>
<td>( \theta = 30 )</td>
<td></td>
</tr>
</tbody>
</table>

\( \text{Area} = 16.75516082 \)

Here, the area of the sector is 24.

Here, the area of the sector is 16.75516082.
Determines the area \( S \) of a segment using the following formula:

\[
S = \left(l - r^2 \sin\left(\frac{l}{r}\right)\right) \frac{1}{2}
\]

(Angle unit = radians)

**OPERATION**

<table>
<thead>
<tr>
<th>5625</th>
<th>(l - r^2 \sin(l/r))/2</th>
<th>l : arc</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>l = 0</td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the area of segment \((l = 30, r = 10)\).

<table>
<thead>
<tr>
<th>30</th>
<th>(l - r^2 \sin(l/r))/2</th>
<th>r : radius</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>r = 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10</th>
<th>(l - r^2 \sin(l/r))/2</th>
<th>r : radius</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Area = 142.9439996</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3</th>
<th>(l - r^2 \sin(l/r))/2</th>
<th>l : arc</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>l = 30</td>
</tr>
</tbody>
</table>

Here, the area of the segment is 142.9439996.
Determines the area (S) of an ellipse using the following formula:

\[ S = \pi ab \]

**OPERATION**

5630 [LB]

\[ xeb \quad a:b:radius \]

\[ b = 0 \ ? - \]

**EXAMPLE**

Determine the area of ellipse (a = 4, b = 6).

4 [EC] 6 [ED]

\[ xeb \quad a:b:radius \]

\[ Area = 75.39822369 \]

\[ xeb \quad a:b:radius \]

\[ b = 4 \ ? - \]

Here, the area of the ellipse is 75.39822369.
Determines the area (S) of a polygon using one of the following formulas:  

1. \( S = f(n, r) = nr^2 \tan \frac{\pi}{n} \)
2. \( S = f(n, R) = \frac{1}{2} nR^2 \sin \frac{2\pi}{n} \)
3. \( S = f(n, l) = \frac{1}{4} nl^2 \cot \frac{\pi}{n} \)  

* \( n \) indicates the number of sides in the polygon. This means that \( n = 6 \) for a regular hexagon.

**OPERATION**

5635  (UB)

**EXAMPLE**

Determine the area of regular hexagon \((r = 5 \ (n = 6))\).

\[
\begin{array}{|c|c|}
\hline
\text{Polygon} & \text{n: number} \\
\hline
\text{n = 6} & \text{r: inside} \\
\hline
\text{Area} & \text{86.60254038} \\
\hline
\text{Area (Polygon)} & \text{1: n, r-A 2: n, R-A 3: n, l-A} \\
\hline
\end{array}
\]

Here, the area of the regular hexagon is 86.60254038.

**EXAMPLE**

Determine the area of regular hexagon \((R = 6)\).

\[
\begin{array}{|c|c|}
\hline
\text{Polygon} & \text{n: number} \\
\hline
\text{n = 6} & \text{r: outside} \\
\hline
\text{Area} & \text{93.53074361} \\
\hline
\text{Area (Polygon)} & \text{1: n, r-A 2: n, R-A 3: n, l-A} \\
\hline
\end{array}
\]

Here, the area of the regular hexagon is 93.53074361.
EXAMPLE

Determine the area of regular hexagon \((n = 4)\).

<table>
<thead>
<tr>
<th>Polygon n: number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(n = 0) ? -</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Polygon ((n = 6)) l: side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area = 41.56921938</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Area (polygon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(l \cdot n \cdot r - A) 2: n \cdot r - A 3: n \cdot l - A</td>
</tr>
</tbody>
</table>

Here, the area of the regular hexagon is 41.56921938.

5650 SURFACE AREA OF A SPHERE

Determines the surface area \((S)\) of a sphere using the following formula:

\[
S = f(n) = 4 \pi r^2
\]

\(r:\) Radius of sphere

OPERATION

5650

<table>
<thead>
<tr>
<th>4 (\pi r^2) r: radius</th>
</tr>
</thead>
</table>

EXAMPLE

Determine the surface area of sphere \(r = 8\).

<table>
<thead>
<tr>
<th>4 (\pi r^2) r: radius</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>8</th>
<th>Surface = 804.2477193</th>
</tr>
</thead>
</table>

Here, the surface area of the sphere is 804.2477193.
Determines the surface area (S) of a zone of a sphere using the following formula:

\[ S = f(r, h, a, b) = 2\pi rh + \pi(a^2 + b^2) \]

**OPERATION**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 EXIT</td>
<td>2( \pi rh + \pi(a^2 + b^2) ) ( a: b: r: \text{radius} ) (Height)</td>
</tr>
<tr>
<td>4 EXIT</td>
<td>2( \pi rh + \pi(a^2 + b^2) ) ( a: b: r: \text{radius} ) (Upper radius)</td>
</tr>
<tr>
<td>5 EXIT</td>
<td>2( \pi rh + \pi(a^2 + b^2) ) ( a: b: r: \text{radius} ) (Lower radius)</td>
</tr>
<tr>
<td>6 EXIT</td>
<td>2( \pi rh + \pi(a^2 + b^2) ) ( a: b: r: \text{radius} ) (Sphere radius)</td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the surface area of zone \( h = 2 \), \( a = 4 \), \( b = 5 \), \( r = 6 \) of a sphere.

\[ S = 2\pi \times 2 + \pi(4^2 + 5^2) = 204.2035225 \]

Here, the surface area of the zone is 204.2035225.
Determines the surface area (S) of a spherical sector using the following formula:

\[ S = f(r, h) = 2\pi rh + \pi ar \]

\[ a = \sqrt{h(2r-h)} \]

**OPERATION**

5660 [LIB]

\[ 2\pi rh + \pi ar = \sqrt{h(2r-h)} \]

\[ r = 0 ? \]

**EXAMPLE**

Determine the surface area of spherical sector (r = 5, h = 3).

5 [EX]

\[ 2\pi rh + \pi ar = \sqrt{h(2r-h)} \]

\[ h = 0 ? \]

3 [EX]

\[ 2\pi rh + \pi ar = \sqrt{h(2r-h)} \]

\[ h = \text{height} \]

\[ \text{Surface} = 166.2307103 \]

[EX]

\[ 2\pi rh + \pi ar = \sqrt{h(2r-h)} \]

\[ r = 5 ? \]

\[ \text{Radius} \]

\[ \text{Height} \]

\[ \text{Return to initial display} \]

Here, the surface area of the spherical sector is 166.2307103.
Determines the surface area (S) of a circular cylinder using the following formula:

\[ S = 2\pi rh + 2\pi r^2 \]

**OPERATION**

**EXAMPLE**

Determine the surface area of a circular cylinder \((r = 6, \ h = 10)\).

<table>
<thead>
<tr>
<th>6</th>
<th>(2\pi rh + 2\pi r^2)</th>
<th>(r): radius</th>
<th>(h): height</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>(2\pi rh + 2\pi r^2)</td>
<td>(r): radius</td>
<td>(h): height</td>
</tr>
</tbody>
</table>

**Surface = 603.1857895**

(Return to initial display)

Here, the surface area of the circular cylinder is 603.1857895.
Determines the surface area (S) of a circular cone using the following formula:

\[ S = \pi r \sqrt{r^2 + h^2} + \pi r^2 \]

**EXAMPLE**

Determine the surface area of circular cone \((r = 6, h = 10)\).

<table>
<thead>
<tr>
<th>( r )</th>
<th>( h )</th>
<th>Calculation</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>h = 0</td>
<td>( r \sqrt{r^2 + h^2} + \pi r^2 )</td>
<td>332.9190432</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>( \pi r \sqrt{r^2 + h^2} + \pi r^2 )</td>
<td>332.9190432</td>
</tr>
</tbody>
</table>

Here, the surface area of the circular cone is 332.9190432.
SURFACE AREA OF A FRUSTUM OF A CIRCULAR CONE

Determines the surface area (S) of a frustum of a circular cone using the following formula:

\[ S = \pi (r + R) \sqrt{h^2 + (R - r)^2} + \pi (R^2 + r^2) \]

**OPERATION**

\[ \text{5675 LIB} \]

\[ \frac{x (R + r)}{r = 0} \frac{\sqrt{h^2 + (R - r)^2} + x (R^2 + r^2)}{? -} \]

**EXAMPLE**

Determine the surface area of the frustum of a circular cone \( r = 4, R = 6, h = 10 \).

4 \ Rex

\[ \frac{x (R + r)}{R = 0} \frac{\sqrt{h^2 + (R - r)^2} + x (R^2 + r^2)}{? -} \] (Upper radius)

6 \ Rex

\[ \frac{x (R + r)}{h = 0} \frac{\sqrt{h^2 + (R - r)^2} + x (R^2 + r^2)}{? -} \] (Lower radius)

10 \ Rex

\[ \frac{x (R + r)}{\text{Surface} = 483.7436629} \frac{\sqrt{h^2 + (R - r)^2} + x (R^2 + r^2)}{r = 4} \] (Height)

\[ \text{5EX} \]

\[ \frac{x (R + r)}{r = 4} \frac{\sqrt{h^2 + (R - r)^2} + x (R^2 + r^2)}{? -} \] (Return to initial display)

Here, the surface area of the frustum of the circular cone is 483.7436629.
Determines the volume \( V \) of a sphere using the following formula:

\[
V = \frac{4}{3} \pi r^3
\]

\( r \) : Radius

**OPERATION**

5700 \( \text{LB} \)

\[
\frac{4 \pi r^3}{3} \quad r: \text{radius}
\]

**EXAMPLE**

Determine the volume of sphere \( r = 6 \).

6 \( \text{EX} \)

\[
\frac{4 \pi r^3}{3} \quad r: \text{radius} \quad \text{(Radius)}
\]

Volume = 904.7786842

6 \( \text{RE} \)

\[
\frac{4 \pi r^3}{3} \quad r: \text{radius} \quad \text{(Return to initial display)}
\]

Here, the volume of the sphere is 904.7786842.
Determine the volume (V) of the zone of a sphere using the following formula:

\[ V = f(a, b, h) = \frac{1}{6} \cdot h \cdot (3a^2 + 3b^2 + h^2) \]

**OPERATION**

5705 [LB]  \[ \frac{x \cdot h \cdot (3a^2 + 3b^2 + h^2)}{6} \quad a : \text{radius} \]

\[ a = 6 \quad ? \]

**EXAMPLE**

Determine the volume of the zone of a sphere (a = 6, b = 4, h = 2).

6 [EX]  \[ \frac{x \cdot h \cdot (3a^2 + 3b^2 + h^2)}{6} \quad a : \text{radius} \]

\[ b = 0 \quad ? \] (Lower radius)

4 [EX]  \[ \frac{x \cdot h \cdot (3a^2 + 3b^2 + h^2)}{6} \quad h : \text{height} \]

\[ h = 0 \quad ? \] (Upper radius)

2 [EX]  \[ \frac{x \cdot h \cdot (3a^2 + 3b^2 + h^2)}{6} \quad h : \text{height} \]

\[ \text{Volume} = 167.5516082 \] (Height)

6 [EX]  \[ \frac{x \cdot h \cdot (3a^2 + 3b^2 + h^2)}{6} \quad a : \text{radius} \]

\[ a = 6 \quad ? \] (Return to initial display)

Here, the volume of the zone of the sphere is 167.5516082.
Determines the volume (V) of a spherical sector using the following formula:

\[ V = f(r, h) = \frac{2}{3} \pi r^2 h \]

**OPERATION**

<table>
<thead>
<tr>
<th>5710</th>
<th>Unit</th>
<th>[ \frac{2}{3} \pi r^2 h ]</th>
<th>r: radius</th>
<th>h: height</th>
</tr>
</thead>
<tbody>
<tr>
<td>r = 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the volume of spherical sector (r = 6, h = 2).

1. \[ \frac{2}{3} \pi r^2 h \]
   - (Radius)
2. \[ \frac{2}{3} \pi r^2 h \]
   - (Height)
3. \[ \frac{2}{3} \pi r^2 h \]
   - (Return to initial display)

Here, the volume of the spherical sector is 150.7964474.
VOLUME OF A CIRCULAR CYLINDER

Determines the volume (V) of a circular cylinder using the following formula:

\[ V = \pi r^2 h \]

OPERATION

5715

```
5715 lb
```

EXAMPLE

Determine the volume of circular cylinder (r = 5, h = 10).

```
<table>
<thead>
<tr>
<th>Ex</th>
<th>r</th>
<th>h</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>5</td>
<td>10</td>
<td>785.3981634</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>10</td>
<td>785.3981634</td>
</tr>
</tbody>
</table>
```

Here, the volume of the circular cylinder is 785.3981634.
Determines the volume (V) of a circular cone using the following formula:

\[ V = \frac{1}{3} \pi r^2 h \]

**OPERATION**

5720

<table>
<thead>
<tr>
<th>[ \frac{1}{3} \pi r^2 h ]</th>
<th>[ r: \text{radius} ]</th>
<th>[ h: \text{height} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ r = 0 ]</td>
<td>[ h = ? ]</td>
<td></td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the volume of circular cone (\( r = 5, \ h = 10 \)).

5

<table>
<thead>
<tr>
<th>[ \frac{1}{3} \pi r^2 h ]</th>
<th>[ r: \text{radius} ]</th>
<th>[ h: \text{height} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ h = 0 ]</td>
<td>[ r = ? ]</td>
<td></td>
</tr>
</tbody>
</table>

10

<table>
<thead>
<tr>
<th>[ \frac{1}{3} \pi r^2 h ]</th>
<th>[ r: \text{radius} ]</th>
<th>[ h: \text{height} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume = 261.7993878</td>
<td>[ r = 5 ]</td>
<td>[ h = ? ]</td>
</tr>
</tbody>
</table>

\( \text{Here, the volume of the circular cone is 261.7993878.} \)
VOLUME OF THE FRUSTUM OF A CIRCULAR CONE

Determines the volume (V) of the frustum of a circular cone using the following formula:

\[ V = \frac{1}{3} \pi h (r^2 + rR + R^2) \]

\( r \): Upper radius
\( R \): Lower radius
\( h \): Height

\[ V = f(r, R, h) = \frac{1}{3} \pi h (r^2 + rR + R^2) \]

**OPERATION**

<table>
<thead>
<tr>
<th>5725 LIB</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{xh(r^2 + rR + R^2)}{3} )</td>
</tr>
<tr>
<td>( r = 0 )</td>
</tr>
<tr>
<td>( R ): radius</td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the volume of the frustum of circular cone (\( r = 4 \), \( R = 6 \), \( h = 10 \)).

<table>
<thead>
<tr>
<th>4 DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{xh(r^2 + rR + R^2)}{3} )</td>
</tr>
<tr>
<td>( R = 0 )</td>
</tr>
<tr>
<td>( r ): radius</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6 DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{xh(r^2 + rR + R^2)}{3} )</td>
</tr>
<tr>
<td>( h = 0 )</td>
</tr>
<tr>
<td>( h ): height</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10 DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{xh(r^2 + rR + R^2)}{3} )</td>
</tr>
<tr>
<td>( V = 795.8701389 )</td>
</tr>
<tr>
<td>( h ): height</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{xh(r^2 + rR + R^2)}{3} )</td>
</tr>
<tr>
<td>( r = 4 )</td>
</tr>
<tr>
<td>( r ): radius</td>
</tr>
</tbody>
</table>

Here, the volume of the frustum of the circular cone is 795.8701389.
VOLUME OF A WEDGE

Determines the volume (V) of a wedge using the following formula:

\[ V = \frac{1}{6}bh(2a+c) \]

**Operation**

<table>
<thead>
<tr>
<th>5730</th>
<th>bh(2a+c)/6</th>
<th>a : b : c : edge</th>
<th>h : height</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a = 0 ?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Example**

Determine the volume of wedge (a = 6, b = 8, c = 4, h = 5).

<table>
<thead>
<tr>
<th>6</th>
<th>bh(2a+c)/6</th>
<th>a : b : c : edge</th>
<th>h : height</th>
<th>(One side a)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>bh(2a+c)/6</td>
<td>a : b : c : edge</td>
<td>h : height</td>
<td>(One side b)</td>
</tr>
<tr>
<td>4</td>
<td>bh(2a+c)/6</td>
<td>a : b : c : edge</td>
<td>h : 0 ?</td>
<td>(One side c)</td>
</tr>
<tr>
<td>5</td>
<td>bh(2a+c)/6</td>
<td>a : b : c : edge</td>
<td>h : 0 ?</td>
<td>(Height)</td>
</tr>
<tr>
<td>6</td>
<td>bh(2a+c)/6</td>
<td>a : b : c : edge</td>
<td>h : height</td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

Here, the volume of the wedge is 106.6666667.
Determines the volume (V) of a pyramid using the following formula:

\[ V = f(a, b, h) = \frac{1}{3}abh \]

**OPERATION**

<table>
<thead>
<tr>
<th>5735</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ a \times b \times h / 3 ]</td>
<td>[ a: \text{edge} \quad b: \text{edge} \quad h: \text{height} ]</td>
</tr>
</tbody>
</table>

**EXAMPLE**

Determine the volume of pyramid (a = 4, b = 5, h = 6).

4 EX 5 EX 6 EX

<table>
<thead>
<tr>
<th>[ a \times b \times h / 3 ]</th>
<th>[ a: \text{edge} \quad b: \text{edge} \quad h: \text{height} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Base dimensions)</td>
<td>(Height)</td>
</tr>
</tbody>
</table>

Here, the volume of the pyramid is 40.
Determines the volume (V) of the frustum of a pyramid using the following formula:

\[ V = \frac{h}{3} (ab + cd + \sqrt{abcd}) \]

**OPERATION**

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5740</td>
<td>Network 1/10</td>
</tr>
<tr>
<td>474</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) e: b: c: d: edge</td>
</tr>
<tr>
<td>674</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) h: edge</td>
</tr>
<tr>
<td>125</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) Volume = 336</td>
</tr>
<tr>
<td>482</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) e: b: c: d: edge</td>
</tr>
<tr>
<td>624</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) h: edge</td>
</tr>
<tr>
<td>656</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) Volume = 336</td>
</tr>
<tr>
<td>482</td>
<td>( h(b + c + \sqrt{abcd}) / 3 ) e: b: c: d: edge</td>
</tr>
</tbody>
</table>

**Example**

Determine the volume of the frustum of pyramid (a = 3, b = 4, c = 6, d = 8, h = 12).

Here, the volume of the frustum of the pyramid is 336.

\[ h = 2 \]
Determines the volume (V) of an ellipsoid using the following formula:

\[ V = \frac{4}{3} \pi abc \]

**OPERATION**

5745 [UB]  
4 x abc / 3  
\[ a : b : c : \text{radius} \]

**EXAMPLE**

Determine the volume of ellipsoid (a = 10, b = 6, c = 5).

<table>
<thead>
<tr>
<th>10 [XEC]</th>
<th>4 x abc / 3</th>
<th>a : b : c : radius</th>
<th>(Radius a)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 [XEC]</td>
<td>4 x abc / 3</td>
<td>a : b : c : radius</td>
<td>(Radius b)</td>
</tr>
<tr>
<td></td>
<td>c = 0 ?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 [XEC]</td>
<td>4 x abc / 3</td>
<td>Volume = 1256.637061</td>
<td>(Radius c)</td>
</tr>
<tr>
<td></td>
<td>a : b : c : radius</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[XEC]</td>
<td>4 x abc / 3</td>
<td>a : b : c : radius</td>
<td>(Return to initial display)</td>
</tr>
<tr>
<td></td>
<td>b = 10 ?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here, the volume of the ellipsoid is 1256.637061.
Determines the radius of the inscribed circle and the circumscribed circle and the length of one side of a polygon from a regular polygon's area.

Angle unit used is the DEG mode.

**OPERATION**

5750 5750

**EXAMPLE**

Determine the radius of the inscribed circle and circumscribed circle and one side of a regular pentagon with an area of 450.

<table>
<thead>
<tr>
<th>450</th>
<th>Polygon (r.R.l) n: number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Enter area A)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>175</th>
<th>Polygon (r.R.l) r: inscribed = 11.12988647</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Radius of inscribed circle displayed when the number of sides of polygon is input)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>175</th>
<th>Polygon (r.R.l) R: outside = 13.75729626</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Display of circumscribed circle radius)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>175</th>
<th>Polygon (r.R.l) l: side = 16.17267171</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Display of one side of pentagon)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>175</th>
<th>Polygon (r.R.l) A: area</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

Here, the radius of the inscribed circle is 11.12988647, the radius of the circumscribed circle 13.75729626, and one side of the regular pentagon is 16.17267171.
Determines four the following parameters for a regular polyhedron when one parameter is input:

- $a$: Length of one side
- $r$: Radius of inscribed sphere
- $R$: Radius of circumscribed sphere
- $S$: Surface area
- $V$: Volume

**OPERATION**

```
  5760 LB
  Select number of face
  1: 4 f 2: 6 f 3: 8 f 4: 12 f 5: 20 f
```

One of the following regular polyhedrons can be selected from the menu illustrated above.

**EXAMPLE**

Find the length of one side ($a$), radius of inscribed sphere ($r$), radius of circumscribed sphere ($R$), and volume ($V$) of a regular octahedron with a surface area of 100 cm$^2$.

```
  3
  Select input data
  1: a 2: r 3: R 4: S 5: V

  Octahedron)
  S: surface
  S = 0 ? _

  Octahedron)
  3.12e-6 v
  S: area = 5.372849659

  Octahedron)
  3.12e-6 v
  (Entry of surface area displays length of one side)

  Octahedron)
  3.12e-6 v
  r: inside = 2.19345688 (Radius of inscribed circle display)

  Octahedron)
  3.12e-6 v
  R: outside = 3.799178428 (Radius of circumscribed circle display)

  Octahedron)
  3.12e-6 v
  V: volume = 73.11522284 (Volume display)

  Octahedron)
  3.12e-6 v
  (Redisplay of radius of circumscribed circle)

  Octahedron)
  3.12e-6 v
  (Redisplay of volume)

  Octahedron)
  3.12e-6 v
  (Return to menu display)
```

```
  4
  Select number of face
  1: 4 f 2: 6 f 3: 8 f 4: 12 f 5: 20 f
```
Here, the following data is calculated for the regular octahedron:

Length of one side : Approximately 5.37cm  
Radius of inscribed circle : Approximately 2.19cm  
Radius of circumscribed circle : Approximately 3.80cm  
Surface area : 100cm²  
Volume : Approximately 73.12cm³

**EXAMPLE**

Find the radius of inscribed sphere (r), radius of circumscribed sphere (R), surface area (S) and volume (V) of a regular icosahedron which has a length of one side of 5cm.

<table>
<thead>
<tr>
<th>Select input data</th>
<th>1 : l</th>
<th>2 : f</th>
<th>3 : R</th>
<th>4 : S</th>
<th>5 : V</th>
</tr>
</thead>
<tbody>
<tr>
<td>(5 : 20f selection from menu)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 : l selected from menu</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 : 3.30e-12v</td>
<td>1.10e+12v</td>
<td>7.752e+00 V</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entry of length of one side displays radius of inscribed circle</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Radius of circumscribed circle display)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 : 4.7552e+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Surface area display)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 : 216.512636</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Volume display)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 : 272.712738</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Return to menu display</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here, the following data is calculated for the regular icosahedron:

Length of one side : 5cm  
Radius of inscribed circle : Approximately 3.78cm  
Radius of circumscribed circle : Approximately 4.76cm  
Surface area : Approximately 216.51cm²  
Volume : Approximately 272.71cm³
Displays the following 23 factorized formulas:

1. \( a^2 - b^2 = (a + b)(a - b) \)
2. \( a^2 \pm b^2 = (a \pm b)(a^2 \mp ab + b^2) \)
3. \( a^4 - b^4 = (a - b)(a + b)(a^2 + b^2) \)
4. \( a^4 + b^4 = (a^2 + \sqrt{2}ab + b^2)(a^2 - \sqrt{2}ab + b^2) \)
5. \( a^2 \pm 2ab + b^2 = (a \pm b)^2 \)
6. \( a^2 \pm 3a^2b + 3ab^2 - b^2 = (a \pm b)^3 \)
7. \( (a \pm b)^2 \mp 4ab = (a \mp b)^2 \)
8. \( a^2 + b^4 + c^2 + 2bc + 2ca + 2ab = (a + b + c)^2 \)
9. \( a^4 + a^2b^2 + b^4 = (a^2 + ab + b^2)(a^2 - ab + b^2) \)
10. \( a^2 + b^2 + c^2 - 3abc = (a + b + c)(a^2 + b^2 + c^2 - bc - ca - ab) \)
11. \( (ac - bd)^2 + (ad + bc)^2 = (a^2 + b^2)(c^2 + d^2) \)
12. \( (ac + bd)^2 + (ad - bc)^2 = (a^2 + b^2)(c^2 + d^2) \)
13. \( (ac + bd)^2 - (ad - bc)^2 = (a^2 - b^2)(c^2 - d^2) \)
14. \( (ac - bd)^2 - (ad + bc)^2 = (a^2 - b^2)(c^2 - d^2) \)
15. \( a^2(b - c) + b^2(c - a) + c^2(a - b) = -(b - c)(c - a)(a - b) \)
16. \( (b - c)^2 + (c - a)^2 + (a - b)^2 = 3(b - c)(c - a)(a - b) \)
17. \( a^2 + b^2 - c^2 - 2bc = (a + b + c)(b - c - a)(a - b - c) \)
18. \( x^2 + (a - b + c)x + ab = (x + a)(x + b) \)
19. \( x^2 + (a - b + c)x + ab = (x + a)(x + b)(x + c) \)
20. \( a^2 - b^2 - c^2 - 2bc + (bc - ca + ab)x + abc = (x + a)(x + b)(x + c) \)
21. \( a^2 - b^2 - c^2 - 2bc = (a + b + c)(b - c - a)(a - b - c) \)
22. \( a^2 - b^2 + c^2 - 3abc = (a + b - c)(a + c)(c + b) \)
23. \( a^2(b - c) - b^2(c - a) - c^2(a - b) = -(b - c)(c - a)(a - b)(a + b + c) \)

**OPERATION**

5800  1B

### (or \( \equiv \)) scrolls to the following formula, \( \equiv \) to the previous formula, \( \equiv \) to the first formula, and \( \equiv \) to the last (23rd) formula.

**EXAMPLE**

Display a desired factorized formula.

\[ \frac{a^2 - b^2}{(a + b)(a - b)} \]

(2) (Formula 2)

\[ \frac{a^2 - 2ab + b^2}{(a - b)^2} \]

(5) (Formula 5)

\[ \frac{a^3 + b^3 + c^3 - 3abc}{(a + b + c)(a + b - c)(b - c)} \]

(10) (Formula 10)

\[ \frac{(a + b)^2 - 4ab}{(a + b)(a - b)} \]

(7) (Formula 7)

\[ \frac{a^2 - b^2}{(a + b)(a - b)} \]

(1) (Formula 1)

\[ \frac{a^3(b - c) + b^3(c - a) + c^3(a - b)}{(b - c)(c - a)(a - b)} \]

(23) (Formula 23)

\[ \frac{x^2 + (a + b + c)x + ab + abc}{x + (a + b)(x + c)} \]

(18) (Formula 19)
Displays the following 38 trigonometric equations:

1. \(\sin^2 \theta + \cos^2 \theta = 1\)
2. \(1 + \tan^2 \theta = \sec^2 \theta\)
3. \(1 + \cot^2 \theta = \csc^2 \theta\)
4. \(\sin(a \pm \beta) = \sin a \cdot \cos b \pm \cos a \cdot \sin b\)
5. \(\cos(a \pm \beta) = \cos a \cdot \cos b \mp \sin a \cdot \sin b\)
6. \(\tan(a \pm \beta) = \frac{\tan a \pm \tan b}{1 \mp \tan a \cdot \tan b}\)
7. \(\cot(a \pm \beta) = \frac{\cot a \mp \cot b}{1 \pm \cot a \cdot \cot b}\)
8. \(\sin 2\theta = 2\sin \theta \cdot \cos \theta\)
9. \(\cos 2\theta = \cos^2 \theta - \sin^2 \theta\)
10. \(\cos 2\theta = 1 - 2\sin^2 \theta\)
11. \(\sin 2\theta = 2\cos \theta \cdot \sin \theta\)
12. \(\tan 2\theta = \frac{2\tan \theta}{1 - \tan^2 \theta}\)
13. \(\sin \frac{\theta}{2} = \pm \sqrt{\frac{1 - \cos \theta}{2}}\)
14. \(\cos \frac{\theta}{2} = \pm \sqrt{\frac{1 + \cos \theta}{2}}\)
15. \(\tan \frac{\theta}{2} = \pm \sqrt{\frac{1 - \cos \theta}{1 + \cos \theta}}\)
16. \(\tan \frac{\theta}{2} = \frac{1 - \cos \theta}{\sin \theta}\)
17. \(\tan \frac{\theta}{2} = \frac{\sin \theta}{1 - \cos \theta}\)
18. \(\cot \frac{\theta}{2} = \cot \theta - \cot \theta\)
19. \(\cot \frac{\theta}{2} = \frac{\cos \theta}{\sin \theta}\)
20. \(\cot \frac{\theta}{2} = \frac{\sin \theta}{\cos \theta}\)
21. \(\cot \frac{\theta}{2} = \frac{1 + \cos \theta}{\sin \theta}\)
22. \(\cot \frac{\theta}{2} = \csc \theta + \cot \theta\)
23. \(\sin 3\theta = 3\sin \theta - 4\sin^3 \theta\)
24. \(\cos 3\theta = 4\cos^3 \theta - 3\cos \theta\)
25. \(\tan 3\theta = \frac{3\tan \theta - \tan^3 \theta}{1 - 3\tan^2 \theta}\)
26. \(2\sin \theta \cdot \cos \theta = \sin(\alpha + \beta) + \sin(\alpha - \beta)\)
27. \(2\cos \theta \cdot \sin \theta = \sin(\alpha + \beta) - \sin(\alpha - \beta)\)
28. \(2\cos^2 \theta = \cos(\alpha + \beta) + \cos(\alpha - \beta)\)
29. \(2\sin^2 \theta = -\cos(\alpha + \beta) - \cos(\alpha - \beta)\)
30. \(\sin(\alpha + \beta) = 2\sin \left(\frac{\alpha + \beta}{2}\right) \cdot \cos \left(\frac{\alpha - \beta}{2}\right)\)
31. \(\sin(\alpha - \beta) = 2\cos \left(\frac{\alpha + \beta}{2}\right) \cdot \sin \left(\frac{\alpha - \beta}{2}\right)\)
32. \(\cos(\alpha + \beta) = 2\cos \left(\frac{\alpha + \beta}{2}\right) \cdot \cos \left(\frac{\alpha - \beta}{2}\right)\)
33. \(\cos(\alpha - \beta) = 2\sin \left(\frac{\alpha + \beta}{2}\right) \cdot \cos \left(\frac{\alpha - \beta}{2}\right)\)
34. \(\tan(45^{\circ} + \theta) = \sec \theta = \tan \theta\)
35. \(\tan(45^{\circ} - \theta) = \frac{1 - \sin \theta}{\cos \theta}\)
36. \(\tan(45^{\circ} + \theta) = \cot(45^{\circ} - \theta)\)
37. \(\tan(45^{\circ} + \theta) = \frac{1 - \tan \theta}{1 + \tan \theta}\)
38. \(\cot(45^{\circ} - \theta) = \frac{1 + \cot \theta}{1 - \cot \theta}\)

**OPERATION**

5810 8 1 2 6 4 6 + - =

(1) [\(\sin^2 \theta + \cos^2 \theta\)]

\[\psi\) (or \(\Theta\)) scrolls to the following equation, \(\Theta\) to the previous equation, \(\Theta\) to the first equation, and \(\Theta\) to the last (38th) equation.

**EXAMPLE**

D) splay a desired trigonometric equation.

\(\psi\)

1. \(1 + \tan^2 \theta\)
2. \(\sec \theta\)
(Equation 2)

\(\psi\) \(\Theta\) \(\Theta\) \(\Theta\)

3. \(\tan(a \pm \beta) = (\tan a \mp \tan b)/(1 \pm \tan a \cdot \tan b)\)
4. \(\sec \theta\)
(Equation 6)

\(\psi\) \(\Theta\) \(\Theta\)

5. \(\cos^2 \theta - \sin^2 \theta\)
6. \(\cos \theta\)
(Equation 9)

\(\psi\) \(\Theta\) \(\Theta\) \(\Theta\)

7. \(\cos a \cdot \cos b\)
8. \(\cos \theta\)
(Equation 5)

\(\psi\) \(\Theta\)

9. \(\sin a \cdot \sin b\)
10. \(\sin \theta\)
(Equation 3)

\(\psi\)

11. \(\cot(45^{\circ} - \theta)\)
12. \(\cot \theta\)
(Equation 38)

\(\psi\) \(\Theta\)

13. \(\cot(45^{\circ} + \theta)\)
14. \(\cot \theta\)
(Equation 1)
Displays the following 38 differential equation:

1. \( y = c \quad y' = 0 \)
2. \( y = x^n \quad y' = nx^{n-1} \)
3. \( y = x \quad y' = 1 \)
4. \( y = \frac{1}{x} \quad y' = -\frac{1}{x^2} \)
5. \( y = \sqrt{x} \quad y' = \frac{1}{2\sqrt{x}} \)
6. \( y = e^x \quad y' = e^x \)
7. \( y = \ln x \quad y' = \frac{1}{x} \)
8. \( y = x \quad y' = x^2 (\log x + 1) \)
9. \( y = \sin x \quad y' = \cos x \)
10. \( y = \cos x \quad y' = -\sin x \)
11. \( y = \tan x \quad y' = \sec^2 x \)
12. \( y = \csc x \quad y' = -\csc^2 x \)
13. \( y = \sec x \quad y' = \tan x \)
14. \( y = \cot x \quad y' = -\csc^2 x \)
15. \( y = \cos x \quad y' = -\sin x \)
16. \( y = \sin x \quad y' = \cos x \cdot \cot x \)
17. \( y = \cos x \quad y' = -\sin x \cdot \cot x \)
18. \( y = \tan x \quad y' = \sec x \cdot \cot x \)
19. \( y = \csc x \quad y' = -\csc^2 x \cdot \cot x \)
20. \( y = \sec x \quad y' = \tan x \cdot \cot x \)
21. \( y = \sin^{-1} x \quad y' = \frac{1}{\sqrt{1 - x^2}} \quad (y < \frac{\pi}{2}) \)
22. \( y = \cos^{-1} x \quad y' = -\frac{1}{\sqrt{1 - x^2}} \quad (0 < y < \pi) \)
23. \( y = \tan^{-1} x \quad y' = \frac{1}{1 + x^2} \quad (y < \frac{\pi}{2}) \)
24. \( y = \cot^{-1} x \quad y' = -\frac{1}{1 + x^2} \quad (y > \frac{\pi}{2}) \)
25. \( y = \sec^{-1} x \quad y' = \frac{1}{x\sqrt{x^2 - 1}} \quad (0 < x, x^2 > 1) \)
26. \( y = \csc^{-1} x \quad y' = \frac{1}{x\sqrt{x^2 - 1}} \quad (y > \frac{\pi}{2}, x^2 > 1) \)
27. \( y = \sin x \quad y' = \cos x \)
28. \( y = \cosh x \quad y' = \sinh x \)
29. \( y = \tanh x \quad y' = \sech^2 x \)
30. \( y = \coth x \quad y' = -\csch^2 x \)
31. \( y = \sech x \quad y' = -\sech x \cdot \tanh x \)
32. \( y = \cosech x \quad y' = -\cosech x \cdot \coth x \)
33. \( y = \sinh x \quad y' = \cosh x \)
34. \( y = \cosh x \quad y' = \sinh x \)
35. \( y = \tanh x \quad y' = \frac{1}{x^2 - 1} \quad (x^2 < 1) \)
36. \( y = \coth x \quad y' = \frac{1}{x^2 - 1} \quad (x^2 > 1) \)
37. \( y = \sech x \quad y' = -\frac{1}{x^2 - 1} \quad (0 < x < 1) \)
38. \( y = \cosech x \quad y' = -\frac{1}{x^2 - 1} \quad (0 < x < 1) \)

**OPERATION**

5820  ![INPUT AND OUTPUT](https://example.com/inputoutput.png)

![EQUATION SCROLL](https://example.com/equationscroll.png)

## EXAMPLE

Display a desired differential equation.

![EQUATION DISPLAY](https://example.com/equation.png)

(Equation 1)

(Equation 2)

(Equation 6)

(Equation 9)

(Equation 12)

(Equation 38)
Displays the following 34 integration equations:

1. $\int dx = x + C$
2. $\int x^2 dx = \frac{x^{n+1}}{n+1} + C \quad (n \neq -1)$
3. $\int \frac{1}{x} dx = \log |x| + C$
4. $\int \frac{1}{x-a} dx = \log |x-a| + C$
5. $\int e^x dx = e^x + C$
6. $\int e^{\alpha x} dx = \frac{1}{\alpha} e^{\alpha x} + C$
7. $\int \alpha x^n dx = \frac{x^{n+1}}{n+1} + C \quad (n > 0, n \neq 1)$
8. $\int e^{\alpha x} dx = \frac{e^{\alpha x}}{\alpha} + C \quad (n > 0, n \neq 1)$
9. $\int \log x dx = (\log x)(x-1) + C$
10. $\int \cos x dx = \frac{1}{2} \sin 2x + C$
11. $\int \sin x dx = -\cos x + C$
12. $\int \frac{1}{\sqrt{a^2 - x^2}} dx = \sin^{-1} \left(\frac{x}{a}\right) + C \quad (x < a)$
13. $\int \frac{1}{\sqrt{a^2 + x^2}} dx = \frac{1}{a} \tan^{-1} \left(\frac{x}{a}\right) + C$
14. $\int \frac{1}{\sqrt{a^2 - x^2}} dx = \frac{1}{a} \log \left|\frac{1}{a} x + \sqrt{x^2 - a^2}\right| + C \quad (x > a)$
15. $\int \frac{1}{\sqrt{a^2 + x^2}} dx = \frac{1}{a} \log \left|\frac{a}{x} + \sqrt{x^2 - a^2}\right| + C$
<table>
<thead>
<tr>
<th>F(p)</th>
<th>f(t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{1}{p} )</td>
<td>1</td>
</tr>
<tr>
<td>( \frac{1}{p^2} )</td>
<td>( t )</td>
</tr>
<tr>
<td>( \frac{1}{p^3} )</td>
<td>( \frac{t^{n-1}}{(n-1)!} ) ( n = 1, 2, 3, \ldots )</td>
</tr>
<tr>
<td>( \frac{1}{p \pm m} )</td>
<td>( e^{\pm at} )</td>
</tr>
<tr>
<td>( \frac{1}{p(p + m)} )</td>
<td>( \frac{1}{m} (1 - e^{-at}) )</td>
</tr>
<tr>
<td>( \frac{1}{p(p + m)} )</td>
<td>( \frac{1}{m} (e^{-at} + mt - 1) )</td>
</tr>
<tr>
<td>( \frac{a}{p^2 + a^2} )</td>
<td>sin ( at )</td>
</tr>
<tr>
<td>( \frac{a}{p^2 + a^2} )</td>
<td>cos ( at )</td>
</tr>
<tr>
<td>( \frac{1}{p^2 + a^2} )</td>
<td>( \frac{1}{a} ) sin ( at )</td>
</tr>
<tr>
<td>( \frac{a}{p^2 - a^2} )</td>
<td>sinh ( at )</td>
</tr>
<tr>
<td>( \frac{p}{p^2 - a^2} )</td>
<td>cosh ( at )</td>
</tr>
<tr>
<td>( \frac{1}{p^2 - a^2} )</td>
<td>( \frac{1}{a} ) sinh ( at )</td>
</tr>
<tr>
<td>( \frac{1}{p(p^2 - a^2)} )</td>
<td>( \frac{1}{a^2} (1 - \cos ( at )) )</td>
</tr>
<tr>
<td>( \frac{1}{p(p^2 - a^2)} )</td>
<td>( \frac{1}{a^3} (at - \sin ( at )) )</td>
</tr>
<tr>
<td>( \frac{1}{(p + m)(p + n)} )</td>
<td>( \frac{1}{n - m} (e^{-at} - e^{-at}) )</td>
</tr>
<tr>
<td>( \frac{1}{(p + m)(p + n)} )</td>
<td>( \frac{1}{m - n} (me^{-at} - ne^{-at}) )</td>
</tr>
<tr>
<td>( \frac{1}{(p + m)^2} )</td>
<td>( t e^{-at} )</td>
</tr>
<tr>
<td>( \frac{1}{(p + m)^2} )</td>
<td>( \frac{1}{(n - 1)!} t^{n-1} e^{-at} ) ( n = 1, 2, 3, \ldots )</td>
</tr>
<tr>
<td>( \frac{p}{(p + m)^3} )</td>
<td>( e^{-at} (1 - mt) )</td>
</tr>
<tr>
<td>( \frac{1}{p(p + m)^3} )</td>
<td>( \frac{1}{m^2} (1 - (1 + mt) e^{-at}) )</td>
</tr>
<tr>
<td>( \frac{1}{p(p + m)^3} )</td>
<td>( \frac{t}{m^3} - \frac{2t}{m^2} e^{-at} - te^{-at} )</td>
</tr>
<tr>
<td>( \frac{p + m}{(p + m)^3} )</td>
<td>( ((n - m)t + 1) e^{-at} )</td>
</tr>
<tr>
<td>( \frac{1}{(p^2 + a^2)^2} )</td>
<td>( \frac{1}{2a} (\sin ( at ) - at \cos ( at )) )</td>
</tr>
<tr>
<td>( F(p) )</td>
<td>( f(t) )</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>( \frac{t}{2a} )</td>
<td>( \frac{t}{2a} \sin at )</td>
</tr>
<tr>
<td>( \frac{t}{2a} )</td>
<td>( \frac{t}{2a} ) ( \sin at + at \cdot \cos at )</td>
</tr>
<tr>
<td>( t \cdot \cos at )</td>
<td>( t \cdot \cos at )</td>
</tr>
<tr>
<td>( \frac{1}{n} e^{-m \cdot \sin ml} )</td>
<td>( \frac{1}{n} e^{-m \cdot \sin ml} )</td>
</tr>
<tr>
<td>( e^{-m \cdot \cos ml} )</td>
<td>( e^{-m \cdot \cos ml} )</td>
</tr>
<tr>
<td>( \frac{1}{2a^4} (\sin at - \sin at) )</td>
<td>( \frac{1}{2a^4} (\sin at - \sin at) )</td>
</tr>
<tr>
<td>( \frac{1}{2a^4} (\cos at - \cos at) )</td>
<td>( \frac{1}{2a^4} (\cos at - \cos at) )</td>
</tr>
<tr>
<td>( \frac{1}{2a^4} (\sin at + \sin at) )</td>
<td>( \frac{1}{2a^4} (\sin at + \sin at) )</td>
</tr>
<tr>
<td>( \frac{1}{2} (\cos at + \cos at) )</td>
<td>( \frac{1}{2} (\cos at + \cos at) )</td>
</tr>
<tr>
<td>( \frac{1}{2a^4} \cdot \sin at \cdot \sin at )</td>
<td>( \frac{1}{2a^4} \cdot \sin at \cdot \sin at )</td>
</tr>
<tr>
<td>( \frac{1}{2a^4} \cdot \cos at \cdot \cos at )</td>
<td>( \frac{1}{2a^4} \cdot \cos at \cdot \cos at )</td>
</tr>
<tr>
<td>( \sin at \cdot \cos at \cdot \sin at )</td>
<td>( \sin at \cdot \cos at \cdot \sin at )</td>
</tr>
<tr>
<td>( -1 + 2e^{-mt} )</td>
<td>( -1 + 2e^{-mt} )</td>
</tr>
<tr>
<td>( \frac{2}{m} - 1 - \frac{2}{m} e^{-mt} )</td>
<td>( \frac{2}{m} - 1 - \frac{2}{m} e^{-mt} )</td>
</tr>
</tbody>
</table>

**Operation**

5840 UB

<table>
<thead>
<tr>
<th>( )</th>
<th>( )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F(p) = \frac{1}{p} )</td>
<td>( 1 )</td>
</tr>
</tbody>
</table>

\( \) or \( \) scrolls to the following equation, \( \) to the previous equation, \( \) to the first equation, and \( \) to the last (36th) equation.

**Example**

Display a desired Laplace transformation equation.

\( \) | \( \) | \( \) | \( \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( F(p) = \frac{1}{p^2} )</td>
<td>( 2 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( F(p) = \frac{e}{(p^2 + 4a^2)} )</td>
<td>( 7 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( F(p) = \frac{1}{(p + m)} )</td>
<td>( 4 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( F(p) = \frac{1}{(p - m) \cdot (p + m)} )</td>
<td>( 6 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( F(p) = \frac{1}{p^2 \cdot (p + m) \cdot (p - m) \cdot (p + m - 1)} )</td>
<td>( 35 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( F(p) = \frac{1}{2} e^{-(s + m - t)} )</td>
<td>( 31 )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| \( F(p) = \frac{1}{p} \) | \( 1 \) | (Equation 1)

261
Periodic Table

Displays the periodic table of elements and atomic weight of selected elements.

- Periodic table of elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Atomic Number</th>
<th>Atomic Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hydrogen</td>
<td>1</td>
<td>1.0107</td>
</tr>
<tr>
<td>Helium</td>
<td>2</td>
<td>4.0026</td>
</tr>
<tr>
<td>Lithium</td>
<td>3</td>
<td>6.941</td>
</tr>
<tr>
<td>Beryllium</td>
<td>4</td>
<td>9.0122</td>
</tr>
<tr>
<td>Boron</td>
<td>5</td>
<td>10.81</td>
</tr>
<tr>
<td>Carbon</td>
<td>6</td>
<td>12.011</td>
</tr>
<tr>
<td>Nitrogen</td>
<td>7</td>
<td>14.0067</td>
</tr>
<tr>
<td>Oxygen</td>
<td>8</td>
<td>15.9945</td>
</tr>
<tr>
<td>Fluorine</td>
<td>9</td>
<td>18.9984</td>
</tr>
<tr>
<td>Neon</td>
<td>10</td>
<td>20.1808</td>
</tr>
<tr>
<td>Sodium</td>
<td>11</td>
<td>22.9898</td>
</tr>
<tr>
<td>Magnesium</td>
<td>12</td>
<td>24.305</td>
</tr>
<tr>
<td>Aluminum</td>
<td>13</td>
<td>26.9815</td>
</tr>
<tr>
<td>Silicon</td>
<td>14</td>
<td>28.0855</td>
</tr>
<tr>
<td>Phosphorus</td>
<td>15</td>
<td>30.9738</td>
</tr>
<tr>
<td>Sulfur</td>
<td>16</td>
<td>32.06</td>
</tr>
<tr>
<td>Chlorine</td>
<td>17</td>
<td>35.453</td>
</tr>
<tr>
<td>Arsenic</td>
<td>18</td>
<td>74.9216</td>
</tr>
<tr>
<td>Antimony</td>
<td>19</td>
<td>121.76</td>
</tr>
<tr>
<td>Tellurium</td>
<td>20</td>
<td>127.60</td>
</tr>
<tr>
<td>Iodine</td>
<td>21</td>
<td>126.9044</td>
</tr>
<tr>
<td>Xenon</td>
<td>22</td>
<td>131.30</td>
</tr>
<tr>
<td>Ruthenium</td>
<td>24</td>
<td>101.07</td>
</tr>
<tr>
<td>Osmium</td>
<td>26</td>
<td>190.23</td>
</tr>
<tr>
<td>Iridium</td>
<td>27</td>
<td>192.22</td>
</tr>
<tr>
<td>Platinum</td>
<td>28</td>
<td>195.08</td>
</tr>
<tr>
<td>Gold</td>
<td>79</td>
<td>196.9665</td>
</tr>
<tr>
<td>Mercury</td>
<td>80</td>
<td>200.59</td>
</tr>
<tr>
<td>Thulium</td>
<td>69</td>
<td>168.932</td>
</tr>
</tbody>
</table>

**Lanthanides elements**

- La
- Ce
- Pr
- Nd
- Pm
- Sm
- Eu
- Gd
- Tb
- Dy
- Ho
- Er
- Tm
- Yb
- Lu

**Actinides elements**

- Ac
- Th
- Pa
- U
- Np
- Pu
- Am
- Cm
- Bk
- Cf
- Es
- Fm
- Md
- No
- Lr

---

The periodic table is a chart that organizes the elements based on their atomic number, atomic weight, and chemical properties. Each element is represented by a symbol, its atomic number, and its atomic weight. The table is divided into periods (rows) and groups (columns) based on the electron configurations of their atoms. Elements in the same group have similar chemical properties, while those in the same period have the same number of electron shells.
## Atomic weight (1)

<table>
<thead>
<tr>
<th>Atomic number</th>
<th>Element</th>
<th>Symbol</th>
<th>Atomic weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hydrogen</td>
<td>H</td>
<td>1.00794 ± 7</td>
</tr>
<tr>
<td>2</td>
<td>Helium</td>
<td>He</td>
<td>4.00260</td>
</tr>
<tr>
<td>3</td>
<td>Lithium</td>
<td>Li</td>
<td>6.941</td>
</tr>
<tr>
<td>4</td>
<td>Beryllium</td>
<td>Be</td>
<td>9.01218</td>
</tr>
<tr>
<td>5</td>
<td>Boron</td>
<td>B</td>
<td>10.81</td>
</tr>
<tr>
<td>6</td>
<td>Carbon</td>
<td>C</td>
<td>12.011</td>
</tr>
<tr>
<td>7</td>
<td>Nitrogen</td>
<td>N</td>
<td>14.0067</td>
</tr>
<tr>
<td>8</td>
<td>Oxygen</td>
<td>O</td>
<td>15.994</td>
</tr>
<tr>
<td>9</td>
<td>Fluorine</td>
<td>F</td>
<td>16.998403</td>
</tr>
<tr>
<td>10</td>
<td>Neon</td>
<td>Ne</td>
<td>20.179</td>
</tr>
<tr>
<td>11</td>
<td>Sodium</td>
<td>Na</td>
<td>22.98977</td>
</tr>
<tr>
<td>12</td>
<td>Magnesium</td>
<td>Mg</td>
<td>24.305</td>
</tr>
<tr>
<td>13</td>
<td>Aluminium</td>
<td>Al</td>
<td>26.98154</td>
</tr>
<tr>
<td>14</td>
<td>Silicon</td>
<td>Si</td>
<td>28.0855</td>
</tr>
<tr>
<td>15</td>
<td>Phosphorus</td>
<td>P</td>
<td>30.97376</td>
</tr>
<tr>
<td>16</td>
<td>Sulfur</td>
<td>S</td>
<td>32.06</td>
</tr>
<tr>
<td>17</td>
<td>Chlorine</td>
<td>Cl</td>
<td>35.453</td>
</tr>
<tr>
<td>18</td>
<td>Argon</td>
<td>Ar</td>
<td>39.946</td>
</tr>
<tr>
<td>19</td>
<td>Potassium</td>
<td>K</td>
<td>39.0983</td>
</tr>
<tr>
<td>20</td>
<td>Calcium</td>
<td>Ca</td>
<td>40.08</td>
</tr>
<tr>
<td>21</td>
<td>Scandium</td>
<td>Sc</td>
<td>44.9559</td>
</tr>
<tr>
<td>22</td>
<td>Titanium</td>
<td>Ti</td>
<td>47.88</td>
</tr>
<tr>
<td>23</td>
<td>Vanadium</td>
<td>V</td>
<td>50.9415</td>
</tr>
<tr>
<td>24</td>
<td>Chromium</td>
<td>Cr</td>
<td>51.996</td>
</tr>
<tr>
<td>25</td>
<td>Manganese</td>
<td>Mn</td>
<td>54.9380</td>
</tr>
</tbody>
</table>
### Atomic weight (2)

<table>
<thead>
<tr>
<th>Atomic number</th>
<th>Element</th>
<th>Symbol</th>
<th>Atomic weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>Iron</td>
<td>Fe</td>
<td>55.847</td>
</tr>
<tr>
<td>27</td>
<td>Cobalt</td>
<td>Co</td>
<td>58.9332</td>
</tr>
<tr>
<td>28</td>
<td>Nickel</td>
<td>Ni</td>
<td>58.69</td>
</tr>
<tr>
<td>29</td>
<td>Copper</td>
<td>Cu</td>
<td>63.546</td>
</tr>
<tr>
<td>30</td>
<td>Zinc</td>
<td>Zn</td>
<td>65.38</td>
</tr>
<tr>
<td>31</td>
<td>Gallium</td>
<td>Ga</td>
<td>69.72</td>
</tr>
<tr>
<td>32</td>
<td>Germanium</td>
<td>Ge</td>
<td>72.59</td>
</tr>
<tr>
<td>33</td>
<td>Arsenic</td>
<td>As</td>
<td>74.9216</td>
</tr>
<tr>
<td>34</td>
<td>Selenium</td>
<td>Se</td>
<td>78.96</td>
</tr>
<tr>
<td>35</td>
<td>Bromine</td>
<td>Br</td>
<td>79.904</td>
</tr>
<tr>
<td>36</td>
<td>Krypton</td>
<td>Kr</td>
<td>83.80</td>
</tr>
<tr>
<td>37</td>
<td>Rubidium</td>
<td>Rb</td>
<td>85.4676</td>
</tr>
<tr>
<td>38</td>
<td>Strontium</td>
<td>Sr</td>
<td>87.62</td>
</tr>
<tr>
<td>39</td>
<td>Yttrium</td>
<td>Y</td>
<td>88.9059</td>
</tr>
<tr>
<td>40</td>
<td>Zirconium</td>
<td>Zr</td>
<td>91.22</td>
</tr>
<tr>
<td>41</td>
<td>Niobium</td>
<td>Nb</td>
<td>92.9064</td>
</tr>
<tr>
<td>42</td>
<td>Molybdenum</td>
<td>Mo</td>
<td>95.94</td>
</tr>
<tr>
<td>43</td>
<td>Technetium</td>
<td>Tc</td>
<td>(98)</td>
</tr>
<tr>
<td>44</td>
<td>Ruthenium</td>
<td>Ru</td>
<td>101.00</td>
</tr>
<tr>
<td>45</td>
<td>Rhodium</td>
<td>Rh</td>
<td>102.9055</td>
</tr>
<tr>
<td>46</td>
<td>Palladium</td>
<td>Pd</td>
<td>106.42</td>
</tr>
<tr>
<td>47</td>
<td>Silver</td>
<td>Ag</td>
<td>107.8682</td>
</tr>
<tr>
<td>48</td>
<td>Cadmium</td>
<td>Cd</td>
<td>112.41</td>
</tr>
<tr>
<td>49</td>
<td>Indium</td>
<td>In</td>
<td>114.82</td>
</tr>
<tr>
<td>50</td>
<td>Tin</td>
<td>Sn</td>
<td>118.69</td>
</tr>
<tr>
<td>51</td>
<td>Antimony</td>
<td>Sb</td>
<td>121.75</td>
</tr>
<tr>
<td>52</td>
<td>Tellurium</td>
<td>Te</td>
<td>127.60</td>
</tr>
<tr>
<td>53</td>
<td>Iodine</td>
<td>I</td>
<td>126.9045</td>
</tr>
<tr>
<td>54</td>
<td>Xenon</td>
<td>Xe</td>
<td>131.29</td>
</tr>
<tr>
<td>55</td>
<td>Caesium</td>
<td>Cs</td>
<td>132.9054</td>
</tr>
<tr>
<td>56</td>
<td>Barium</td>
<td>Ba</td>
<td>137.33</td>
</tr>
<tr>
<td>57</td>
<td>Lanthanum</td>
<td>La</td>
<td>138.9055</td>
</tr>
<tr>
<td>58</td>
<td>Cerium</td>
<td>Ce</td>
<td>140.12</td>
</tr>
<tr>
<td>59</td>
<td>Praseodymium</td>
<td>Pr</td>
<td>140.9077</td>
</tr>
<tr>
<td>60</td>
<td>Neodymium</td>
<td>Nd</td>
<td>144.24</td>
</tr>
<tr>
<td>61</td>
<td>Promethium</td>
<td>Pm</td>
<td>(145)</td>
</tr>
<tr>
<td>62</td>
<td>Samarium</td>
<td>Sm</td>
<td>150.34</td>
</tr>
<tr>
<td>63</td>
<td>Europium</td>
<td>Eu</td>
<td>151.96</td>
</tr>
<tr>
<td>64</td>
<td>Gadolinium</td>
<td>Gd</td>
<td>157.25</td>
</tr>
<tr>
<td>65</td>
<td>Terbium</td>
<td>Tb</td>
<td>158.9254</td>
</tr>
<tr>
<td>Atomic number</td>
<td>Element</td>
<td>Symbol</td>
<td>Atomic weight</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
<td>--------</td>
<td>---------------</td>
</tr>
<tr>
<td>66</td>
<td>Dysprosium</td>
<td>Dy</td>
<td>162.56</td>
</tr>
<tr>
<td>67</td>
<td>Holmium</td>
<td>Ho</td>
<td>164.9304</td>
</tr>
<tr>
<td>68</td>
<td>Erbium</td>
<td>Er</td>
<td>167.25</td>
</tr>
<tr>
<td>69</td>
<td>Thulium</td>
<td>Tm</td>
<td>168.9342</td>
</tr>
<tr>
<td>70</td>
<td>Ytterbium</td>
<td>Yb</td>
<td>173.04</td>
</tr>
<tr>
<td>71</td>
<td>Lutetium</td>
<td>Lu</td>
<td>174.967</td>
</tr>
<tr>
<td>72</td>
<td>Hafnium</td>
<td>Hf</td>
<td>178.49</td>
</tr>
<tr>
<td>73</td>
<td>Tantalum</td>
<td>Ta</td>
<td>180.9479</td>
</tr>
<tr>
<td>74</td>
<td>Tungsten</td>
<td>W</td>
<td>183.84</td>
</tr>
<tr>
<td>75</td>
<td>Rhenium</td>
<td>Re</td>
<td>186.207</td>
</tr>
<tr>
<td>76</td>
<td>Osmium</td>
<td>Os</td>
<td>190.2</td>
</tr>
<tr>
<td>77</td>
<td>Iridium</td>
<td>Ir</td>
<td>192.22</td>
</tr>
<tr>
<td>78</td>
<td>Platinum</td>
<td>Pt</td>
<td>195.08</td>
</tr>
<tr>
<td>79</td>
<td>Gold</td>
<td>Au</td>
<td>196.9665</td>
</tr>
<tr>
<td>80</td>
<td>Mercury</td>
<td>Hg</td>
<td>200.59</td>
</tr>
<tr>
<td>81</td>
<td>Thallium</td>
<td>Tl</td>
<td>204.383</td>
</tr>
<tr>
<td>82</td>
<td>Lead</td>
<td>Pb</td>
<td>207.2</td>
</tr>
<tr>
<td>83</td>
<td>Bismuth</td>
<td>Bi</td>
<td>208.9804</td>
</tr>
<tr>
<td>84</td>
<td>Polonium</td>
<td>Po</td>
<td>(209)</td>
</tr>
<tr>
<td>85</td>
<td>Astatine</td>
<td>At</td>
<td>(210)</td>
</tr>
<tr>
<td>86</td>
<td>Radon</td>
<td>Rn</td>
<td>(222)</td>
</tr>
<tr>
<td>87</td>
<td>Francium</td>
<td>Fr</td>
<td>(223)</td>
</tr>
<tr>
<td>88</td>
<td>Radium</td>
<td>Ra</td>
<td>226.0254</td>
</tr>
<tr>
<td>89</td>
<td>Actinium</td>
<td>Ac</td>
<td>227.0278</td>
</tr>
<tr>
<td>90</td>
<td>Thorium</td>
<td>Th</td>
<td>232.0381</td>
</tr>
<tr>
<td>91</td>
<td>Protactinium</td>
<td>Pa</td>
<td>231.0359</td>
</tr>
<tr>
<td>92</td>
<td>Uranium</td>
<td>U</td>
<td>238.0289</td>
</tr>
<tr>
<td>93</td>
<td>Neptunium</td>
<td>Np</td>
<td>237.0482</td>
</tr>
<tr>
<td>94</td>
<td>Plutonium</td>
<td>Pu</td>
<td>(244)</td>
</tr>
<tr>
<td>95</td>
<td>Americium</td>
<td>Am</td>
<td>(243)</td>
</tr>
<tr>
<td>96</td>
<td>Curium</td>
<td>Cm</td>
<td>(247)</td>
</tr>
<tr>
<td>97</td>
<td>Berkelium</td>
<td>Bk</td>
<td>(247)</td>
</tr>
<tr>
<td>98</td>
<td>Californium</td>
<td>Cf</td>
<td>(251)</td>
</tr>
<tr>
<td>99</td>
<td>Einsteinium</td>
<td>Es</td>
<td>(252)</td>
</tr>
<tr>
<td>100</td>
<td>Fermium</td>
<td>Fm</td>
<td>(257)</td>
</tr>
<tr>
<td>101</td>
<td>Mendelevium</td>
<td>Md</td>
<td>(258)</td>
</tr>
<tr>
<td>102</td>
<td>Nobellium</td>
<td>No</td>
<td>(259)</td>
</tr>
<tr>
<td>103</td>
<td>Lawrencium</td>
<td>Lr</td>
<td>(260)</td>
</tr>
</tbody>
</table>
OPERATION

5900 LB

1 a 2 a 3 a 4 a 5 a 6 a 7 a 8 B B B
1 H

Pressing [X] displays the following periodic element, while pressing [D] displays the previous periodic element. Pressing [C] displays groups 1a-8, while [E] displays groups 1b-7b and 0. Pressing [C] enters input stand by, during which inputting a symbol of an element displays its atomic weight.

EXAMPLE

Display the periodic table at a specific location and display the atomic weight of silicon.

[C] 1 a 2 a 3 a 4 a 5 a 6 a 7 a 8 B B B (#2 periodic element)
[C] 1 a 2 a 3 a 4 a 5 a 6 a 7 a B B B (#5 periodic element)
[C] 5 Rb Sr Y Zr Nb Mo Tc Ru Rh Pd (Group 1b-7b and 0 with same period)
[C] 5 Ag Cd In Sn Sb Te I Xe (Lanthanoid)
[C] 6 La Ce Pr Nd Sm Eu Gd Tb Dy Ho Er Tm Yb Lu (Actinoid)
[C] 7 Actinoid Ac Th Pa U Na Pu Am Cm Bk Cf Es Fm Md No Lr (Enter Si to display atomic weight of silicon)

Input the symbol: Si

Input the symbol: Si Silicon 14 (3-4b) 28.0855

(Enter to initial display)
### Scientific Constants

Displays the following 22 scientific constants. Alphabet keys A–Z can be used to assign displayed values to numeric variables A through Z.

<table>
<thead>
<tr>
<th>Name &amp; Symbol</th>
<th>Value</th>
<th>Unit SI</th>
<th>Unit CGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faraday constant</td>
<td>F</td>
<td>9.64856 × 10^4 C·mol⁻¹</td>
<td>10^3 emu·mol⁻¹</td>
</tr>
<tr>
<td>Gravitational constant</td>
<td>G</td>
<td>6.6720 × 10⁻¹ kg·m⁻¹·s⁻²</td>
<td>10⁻⁸ cm³·s⁻²·g⁻¹</td>
</tr>
<tr>
<td>Avogadro constant</td>
<td>Nₐ</td>
<td>6.022045 mol⁻¹</td>
<td>10^23 mol⁻¹</td>
</tr>
<tr>
<td>Solar gas constant</td>
<td>Rₚ</td>
<td>8.31441 J·mol⁻¹·K⁻¹</td>
<td>10² erg·mol⁻¹·K⁻¹</td>
</tr>
<tr>
<td>Rydberg constant</td>
<td>R₢</td>
<td>1.097373177 × 10⁷ m⁻¹</td>
<td>10⁵ cm⁻¹</td>
</tr>
<tr>
<td>Solar volume of ideal gas at s.t.p.</td>
<td>Vₚ</td>
<td>22.41383 m³·mol⁻¹</td>
<td>10³ cm³·mol⁻¹</td>
</tr>
<tr>
<td>Astron radius</td>
<td>a</td>
<td>5.2917706 m</td>
<td>10⁻⁶ cm</td>
</tr>
<tr>
<td>Speed of light in vacuum</td>
<td>c</td>
<td>299792458 m·s⁻¹</td>
<td>10¹⁶ cm·s⁻¹</td>
</tr>
<tr>
<td>Elementary charge</td>
<td>e</td>
<td>1.6021892 C</td>
<td>10⁻¹⁹ emu</td>
</tr>
<tr>
<td>Gravitational acceleration</td>
<td>g</td>
<td>9.80665 m·s²⁻¹</td>
<td>10² cm·s²</td>
</tr>
<tr>
<td>Planck constant</td>
<td>h</td>
<td>6.626176 × 10⁻³ J·s</td>
<td>10⁻²⁷ erg·s</td>
</tr>
<tr>
<td>Boltzmann constant</td>
<td>k</td>
<td>1.3806462 K</td>
<td>10⁻⁴ erg·K</td>
</tr>
<tr>
<td>Electron rest mass</td>
<td>mₑ</td>
<td>9.109334 kg</td>
<td>10⁻²³ g</td>
</tr>
<tr>
<td>Electron rest mass</td>
<td>mₑ</td>
<td>1.6726485 kg</td>
<td>10⁻²⁷ g</td>
</tr>
<tr>
<td>Proton rest mass</td>
<td>m_p</td>
<td>1.6726485 kg</td>
<td>10⁻²⁷ g</td>
</tr>
<tr>
<td>Atomic mass unit</td>
<td>u</td>
<td>1.6605388 kg</td>
<td>10⁻²⁷ g</td>
</tr>
<tr>
<td>Speed of light in vacuum</td>
<td>c</td>
<td>8.854187813 × 10⁻¹ m·s⁻¹</td>
<td>10⁻¹ⁱ F·m⁻¹</td>
</tr>
<tr>
<td>Magnetic permeability of vacuum</td>
<td>µ₀</td>
<td>1.25663706144 × 10⁻⁷ H·m⁻¹</td>
<td></td>
</tr>
<tr>
<td>Nuclear magneton</td>
<td>µₙ</td>
<td>9.274078 J·T⁻¹</td>
<td>10⁻²¹ erg·G⁻¹</td>
</tr>
<tr>
<td>Magnetic moment</td>
<td>µₑ</td>
<td>9.264332 J·T⁻¹</td>
<td>10⁻²¹ erg·G⁻¹</td>
</tr>
<tr>
<td>Magnetic moment</td>
<td>µₚ</td>
<td>1.1406171 J·T⁻¹</td>
<td>10⁻²³ erg·G⁻¹</td>
</tr>
<tr>
<td>Planck-Boltzmann's constant</td>
<td>σ</td>
<td>5.67032 W·m⁻²·K⁻¹</td>
<td>10⁻⁵ erg·s⁻¹·K⁻¹</td>
</tr>
</tbody>
</table>

The values of these scientific constants are based on JIS Z-8202-1978 (JIS = Japan Industrial Standard).

#### OPERATION

- **5910** displays the following constant, while pressing **0** displays the previous constant. Pressing **D** displays in SI units, while pressing **E** displays in CGS unit. Alphabet keys A–Z can be used to assign displayed values to numeric variables A through Z.

- **5911**
**EXAMPLE**

Display the molar volume of ideal gas at s.t.p. and assign the value to a numeric variable \( V \) in CGS units. Then display the Avogadro constant and assign the value to numeric variable \( N \).

<table>
<thead>
<tr>
<th>V</th>
<th>( V = 22.41383 \times 10^{-3} ) ( \text{m}^3 \cdot \text{mol}^{-1} )</th>
<th>(Molar volume of ideal gas at s.t.p.)</th>
<th>(Displayed in CGS units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>( N = 6.022045 \times 10^{23} ) ( \text{mol}^{-1} )</td>
<td>(Value assigned to numeric variable ( V ))</td>
<td>(Avogadro constant)</td>
</tr>
<tr>
<td>N</td>
<td>( N = 6.022045 \times 10^{23} ) ( \text{mol}^{-1} )</td>
<td>(Value assigned to numeric variable ( N ))</td>
<td>(Exit currently specified constant)</td>
</tr>
</tbody>
</table>

* Constants assigned to numeric variables are retained even when power is switched OFF. Numeric variables such as \( N \) and \( V \) can be used in BASIC programs.
ELECTROLYTIC DISSOCIATION CONSTANTS

Displays the following eight ionization equilibrium formulas:

<table>
<thead>
<tr>
<th>IONIZATION EQUILIBRIUM FORMULAS OF ACID</th>
<th>IONIZATION EQUILIBRIUM CONSTANT $K_a$ (mol/l)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{HCOOH} = \text{HCOO}^- + \text{H}^+$</td>
<td>$1.77 \times 10^{-4}$</td>
</tr>
<tr>
<td>$\text{CH}_2\text{COOH} = \text{CH}_2\text{COO}^- + \text{H}^+$</td>
<td>$1.75 \times 10^{-5}$</td>
</tr>
<tr>
<td>$\text{C}_6\text{H}_5\text{COOH} = \text{C}_6\text{H}_5\text{COO}^- + \text{H}^+$</td>
<td>$6.31 \times 10^{-5}$</td>
</tr>
<tr>
<td>$\text{H}_2\text{CO}_3 = \text{H}^+ + \text{HCO}_3^-$</td>
<td>$4.45 \times 10^{-7}$</td>
</tr>
<tr>
<td>$\text{C}_6\text{H}_5\text{OH} = \text{C}_6\text{H}_5\text{O}^- + \text{H}^+$</td>
<td>$1.00 \times 10^{-7}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IONIZATION EQUILIBRIUM FORMULAS OF BASE</th>
<th>ELECTROLYTIC DISSOCIATION CONSTANT $K_b$ (mol/l)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{NH}_3 + \text{H}_2\text{O} = \text{NH}_4^+ + \text{OH}^-$</td>
<td>$1.78 \times 10^{-5}$</td>
</tr>
<tr>
<td>$\text{C}_6\text{H}_5\text{NH}_2 + \text{H}_2\text{O} = \text{C}_6\text{H}_5\text{NH}_3^+ + \text{OH}^-$</td>
<td>$1.5 \times 10^{-9}$</td>
</tr>
<tr>
<td>$\text{C}_6\text{H}_5\text{NH}_3^+ + \text{H}_2\text{O} = \text{C}_6\text{H}_5\text{NH}_4^+$</td>
<td>$3.6 \times 10^{-10}$</td>
</tr>
</tbody>
</table>

PERATION

5920 UB $\text{HCODH} = \text{HCOO}^- + \text{H}^+$ $K_b = 1.77 \times 10^{-4}$ (mol/l) (1)

$\leftarrow$ (or $\rightarrow$ ) scrolls to the following formula, $\downarrow$ to the previous formula, $\uparrow$ to the first formula, and $\rightarrow$ to the last (8th) formula.

EXAMPLE

Display a desired ionization equilibrium formula.

$\uparrow$

$\text{CH}_3\text{CODM} = \text{CH}_3\text{COO}^- + \text{H}^+$ $K_b = 1.75 \times 10^{-5}$ (mol/l) (2)

$\rightarrow$

$\text{C}_6\text{H}_5\text{OH} = \text{C}_6\text{H}_5\text{O}^- + \text{H}^+$ $K_b = 1.00 \times 10^{-7}$ (mol/l) (5)

$\rightarrow$

$\text{C}_6\text{H}_5\text{NH}_2 + \text{H}_2\text{O} = \text{C}_6\text{H}_5\text{NH}_3^+ + \text{OH}^-$ $K_b = 3.8 \times 10^{-10}$ (mol/l) (8)

$\rightarrow$

$\text{NH}_3 + \text{H}_2\text{O} = \text{NH}_4^+ + \text{OH}^-$ $K_b = 1.78 \times 10^{-9}$ (mol/l) (6)

$\rightarrow$

$\text{HCOOH} = \text{HCOO}^- + \text{H}^+$ $K_b = 1.77 \times 10^{-4}$ (mol/l) (1)
**MOTION AND ENERGY**

Displays the following 20 scientific formulas:

<table>
<thead>
<tr>
<th>NAME</th>
<th>FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniformly accelerated motion</td>
<td>( v = v_0 + at, \ a = \frac{4v}{\Delta t}, \ s = v_0t + \frac{1}{2}at^2 )</td>
</tr>
<tr>
<td>Newton's equation of motion</td>
<td>( F = ma )</td>
</tr>
<tr>
<td>Circular motion (1)</td>
<td>( T = \frac{2\pi}{\omega} = \frac{2\pi}{\omega} = \frac{1}{\omega} )</td>
</tr>
<tr>
<td>Circular motion (2)</td>
<td>( \omega = \frac{2\pi}{T} = 2\pi f = \frac{v}{r} ), ( F = m\omega^2 = \frac{mv^2}{r} )</td>
</tr>
<tr>
<td>Simple harmonic oscillation</td>
<td>( x = r \cdot \sin \omega t, \ v = r \cdot \omega \cdot \cos \omega t, \ a = -\omega^2 x )</td>
</tr>
<tr>
<td>Hooke's law</td>
<td>( F = -kx )</td>
</tr>
<tr>
<td>Spring oscillation</td>
<td>( a = F/m = -\frac{k}{m} x, \ T = 2\pi \sqrt{\frac{m}{k}} )</td>
</tr>
<tr>
<td>Simple pendulum</td>
<td>( a = F/m = -\frac{R}{l} x, \ T = 2\pi \sqrt{\frac{l}{g}} )</td>
</tr>
<tr>
<td>Potential energy (spring)</td>
<td>( E_p = mgh )</td>
</tr>
<tr>
<td>Elastic energy</td>
<td>( E_e = \frac{1}{2}kx^2 )</td>
</tr>
<tr>
<td>Kinetic energy</td>
<td>( E_k = \frac{1}{2}mv^2 )</td>
</tr>
<tr>
<td>Coefficient of friction</td>
<td>( F = \mu N )</td>
</tr>
<tr>
<td>Work</td>
<td>( W = Fs )</td>
</tr>
<tr>
<td>Kepler's law</td>
<td>( T^2/r^3 = \text{Constant} )</td>
</tr>
<tr>
<td>Universal gravitation</td>
<td>( F = G \frac{Mm}{r^2}, \ G = 6.7 \times 10^{-11} \text{N} \cdot \text{m}^2/\text{kg}^2 )</td>
</tr>
<tr>
<td>Potential energy (interplanetary)</td>
<td>( U_p = -G \frac{Mm}{r} )</td>
</tr>
<tr>
<td>Kinetic energy (interplanetary)</td>
<td>( E_k = \frac{1}{2}mr^2 \omega^2 )</td>
</tr>
<tr>
<td>Moment of inertia</td>
<td>( I = mr^2, \ E = \frac{1}{2}I \omega^2 )</td>
</tr>
<tr>
<td>Angular momentum</td>
<td>( J = I \omega )</td>
</tr>
<tr>
<td>Conservation of momentum</td>
<td>( mv_1 + MV_1 = mv_2 + MV_2 )</td>
</tr>
</tbody>
</table>

**NOTE:** Universal gravitational constant displayed as rounded value (see 5910 for details)

**OPERATION**

5930 [LB]  
**Uniformly accelerated motion (1)**  
\( v = v_0 + at, \ a = \frac{4v}{\Delta t}, \ s = v_0t + \frac{1}{2}at^2 \)

[LB] (or [RB]) scrolls to the following formula, [L] to the previous formula, [R] to the first formula, and [C] to the last (20th) formula.
**EXAMPLE**

Display a desired scientific formula.

<table>
<thead>
<tr>
<th>Newton's equation of motion [2]</th>
<th>(Formula 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hook's law [6] [ F = -ky ]</td>
<td>(Formula 6)</td>
</tr>
<tr>
<td>Conservation of momentum [20]</td>
<td>(Formula 20)</td>
</tr>
<tr>
<td>[mv_1 + MV_1 = mv_2 + MV_2]</td>
<td></td>
</tr>
<tr>
<td>Kinetic energy (planet) [17]</td>
<td>(Formula 17)</td>
</tr>
<tr>
<td>[E_k = \frac{1}{2}mv^2]</td>
<td></td>
</tr>
<tr>
<td>Coefficient of friction [12]</td>
<td>(Formula 12)</td>
</tr>
<tr>
<td>[F = \mu N]</td>
<td></td>
</tr>
<tr>
<td>Uniformly accelerated motion [1]</td>
<td>(Formula 1)</td>
</tr>
<tr>
<td>[v = v_0 + at] [s = \frac{1}{2}at^2] [s = v_0t + \frac{1}{2}at^2]</td>
<td></td>
</tr>
</tbody>
</table>
## Wave Motion

Displays the following 16 scientific formulas:

<table>
<thead>
<tr>
<th>NAME</th>
<th>FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wave</td>
<td>( v = \frac{A}{T} = f \lambda, y = a \cdot \sin \omega (t - \frac{x}{\lambda}) )</td>
</tr>
<tr>
<td>Velocity of transverse wave on a string</td>
<td>( v = \sqrt{\frac{F}{\rho}} )</td>
</tr>
<tr>
<td>Interference</td>
<td>( l_{2n+1} = (2n+1) \frac{\lambda}{2}, l_{2n-1} = n\lambda )</td>
</tr>
<tr>
<td>Stationary wave</td>
<td>( l = n \frac{\lambda}{2}, l_{1} = (2n-1) \frac{\lambda}{4} \quad (n = 0) )</td>
</tr>
<tr>
<td>Refraction of wave</td>
<td>( n = \sin \theta'/\sin \theta = v_1/v_2 = \lambda_2/\lambda_1 )</td>
</tr>
<tr>
<td>Natural frequency</td>
<td>( f = \frac{1}{2} \sqrt{\frac{T}{\rho}} )</td>
</tr>
<tr>
<td>Velocity of sound</td>
<td>( v = 331.5 + 0.6T )</td>
</tr>
<tr>
<td>Doppler effect</td>
<td>( f = f_0 \frac{v - v_1}{v - v_2} )</td>
</tr>
<tr>
<td>Beat</td>
<td>( f = f_1 - f_2 \quad (f_1 &gt; f_2) )</td>
</tr>
<tr>
<td>Reflectivity of light</td>
<td>( R_0 = \left( \frac{n_1 - n_2}{n_1 + n_2} \right)^2 )</td>
</tr>
<tr>
<td>Critical angle</td>
<td>( \sin \theta = \frac{n_1}{n_2} )</td>
</tr>
<tr>
<td>De Broglie wave</td>
<td>( \lambda = \frac{h}{mv} )</td>
</tr>
<tr>
<td>Quantum condition</td>
<td>( 2\pi r = \frac{nh}{mv} = n\lambda )</td>
</tr>
<tr>
<td>Photoelectric effect</td>
<td>( \frac{1}{2} mv^2 = h\nu - W )</td>
</tr>
<tr>
<td>Frequency condition</td>
<td>( h\nu = Em - En \quad (m &gt; n) )</td>
</tr>
<tr>
<td>Light wave</td>
<td>( \lambda = c/\nu, c = 2.998 \times 10^8 \text{ (m/s)} )</td>
</tr>
</tbody>
</table>

**OPERATION**

**5932**

**5932**

(1) \( v = \frac{\lambda}{T} = f \lambda \), \( y = a \cdot \sin \omega (t - \frac{x}{\lambda}) \)

(2) \( v = \sqrt{\frac{F}{\rho}} \)

(3) \( n = \sin \theta'/\sin \theta = v_1/v_2 = \lambda_2/\lambda_1 \)

(4) \( f = f_0 \frac{v - v_1}{v - v_2} \)

(5) \( f = f_1 - f_2 \quad (f_1 > f_2) \)

(6) \( R_0 = \left( \frac{n_1 - n_2}{n_1 + n_2} \right)^2 \)

(7) \( \sin \theta = \frac{n_1}{n_2} \)

(8) \( \lambda = \frac{h}{mv} \)

(9) \( 2\pi r = \frac{nh}{mv} = n\lambda \)

(10) \( \frac{1}{2} mv^2 = h\nu - W \)

(11) \( h\nu = Em - En \quad (m > n) \)

(12) \( \lambda = c/\nu, c = 2.998 \times 10^8 \text{ (m/s)} \)

(13) \( v = \frac{\lambda}{T} = f \lambda \), \( y = a \cdot \sin \omega (t - \frac{x}{\lambda}) \)

(14) \( v = \sqrt{\frac{F}{\rho}} \)

(15) \( n = \sin \theta'/\sin \theta = v_1/v_2 = \lambda_2/\lambda_1 \)

(16) \( f = f_0 \frac{v - v_1}{v - v_2} \)

(17) \( f = f_1 - f_2 \quad (f_1 > f_2) \)

(18) \( R_0 = \left( \frac{n_1 - n_2}{n_1 + n_2} \right)^2 \)

(19) \( \sin \theta = \frac{n_1}{n_2} \)

(20) \( \lambda = \frac{h}{mv} \)

(21) \( 2\pi r = \frac{nh}{mv} = n\lambda \)

(22) \( \frac{1}{2} mv^2 = h\nu - W \)

(23) \( h\nu = Em - En \quad (m > n) \)

(24) \( \lambda = c/\nu, c = 2.998 \times 10^8 \text{ (m/s)} \)

**EXAMPLE**

Display a desired wave formula.

(1) \( v = \frac{\lambda}{T} = f \lambda \), \( y = a \cdot \sin \omega (t - \frac{x}{\lambda}) \)

(2) \( v = \sqrt{\frac{F}{\rho}} \)

(5) \( n = \sin \theta'/\sin \theta = v_1/v_2 = \lambda_2/\lambda_1 \)

(16) \( f = f_0 \frac{v - v_1}{v - v_2} \)

(12) \( \lambda = c/\nu, c = 2.998 \times 10^8 \text{ (m/s)} \)

(1) \( v = \frac{\lambda}{T} = f \lambda \), \( y = a \cdot \sin \omega (t - \frac{x}{\lambda}) \)
Displays the following 16 scientific formulas:

<table>
<thead>
<tr>
<th>NAME</th>
<th>FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ohm's law</td>
<td>( V=IR \quad (I=\frac{Q}{t}, R=\rho \cdot \frac{1}{S}) )</td>
</tr>
<tr>
<td>Electric resistance (parallel, series)</td>
<td>( R=R_1+R_2, \quad \frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} )</td>
</tr>
<tr>
<td>DC circuit</td>
<td>( V=E-IR )</td>
</tr>
<tr>
<td>DC power and Joule heat</td>
<td>( P=IV=IP, \quad W=IVt=Pt )</td>
</tr>
<tr>
<td>Conductance</td>
<td>( G=\frac{1}{R} = \frac{1}{V} )</td>
</tr>
<tr>
<td>Kirchhoff's law</td>
<td>( \sum ±V=0, \quad \sum ±I=0 )</td>
</tr>
<tr>
<td>Wheatstone bridge</td>
<td>( R_oR_1 = R_oR_2 )</td>
</tr>
<tr>
<td>Instantaneous value (AC voltage and current)</td>
<td>( V=V_0\sin \omega t, \quad I=I_0\sin \omega t )</td>
</tr>
<tr>
<td>Effective value</td>
<td>( I=\frac{I_0}{\sqrt{2}}, \quad V=\frac{V_0}{\sqrt{2}} )</td>
</tr>
<tr>
<td>AC power</td>
<td>( P=\frac{V_1I_1}{2} )</td>
</tr>
<tr>
<td>Power factor</td>
<td>( P=VI\cos \phi )</td>
</tr>
<tr>
<td>Transformer</td>
<td>( I_1V_1=I_2V_2, \quad \frac{N_2}{N_1} = \frac{V_2}{V_1} )</td>
</tr>
<tr>
<td>Reactance</td>
<td>( X=\omega L=2\pi fL, \quad X=\frac{1}{\omega C} = \frac{1}{2\pi fC} )</td>
</tr>
<tr>
<td>Impedance</td>
<td>( Z=\sqrt{R^2+(\omega L-\frac{1}{\omega C})^2}, \quad V_0=ZI_0 )</td>
</tr>
<tr>
<td>Natural frequency (Natural oscillation)</td>
<td>( f_0=\frac{1}{2\pi\sqrt{LC}} )</td>
</tr>
<tr>
<td>Electric oscillation</td>
<td>( \frac{1}{2}\frac{Q^2}{C} + \frac{1}{2}LI^2=\text{Constant} )</td>
</tr>
</tbody>
</table>

**OPERATION**

### 5934 UB

1. Enter the formula you need to calculate.
2. Enter the value(s) for each variable.
3. Display the result.
4. (or ) scrolls to the following formula.
5. (or ) scrolls to the previous formula.
6. (or ) scrolls to the first formula.
7. (or ) scrolls to the last (16th) formula.

**EXAMPLE**

Display a desired electrical formula.

- **Resistance**
  \( R=R_1+R_2, \quad \frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} \) (Formula 2)
- **Kirchhoff's law**
  \( I=I_1=0, \quad I=V=0 \) (Formula 6)
- **Electric oscillation**
  \( \frac{1}{2}\frac{Q^2}{C} + \frac{1}{2}LI^2=\text{Constant} \) (Formula 16)
- **Reactance**
  \( X=\omega L=2\pi fL, \quad X=\frac{1}{\omega C} = \frac{1}{2\pi fC} \) (Formula 13)
- **Power factor**
  \( P=VI\cos \phi \) (Formula 11)
- **Ohm's law**
  \( V=IR \quad (I=\frac{Q}{t}, R=\rho \cdot \frac{1}{S}) \) (Formula 1)
**ELECTRIC AND MAGNETIC FIELDS**

Displays the following 17 scientific formulas:

<table>
<thead>
<tr>
<th>NAME</th>
<th>FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coulomb's law (Electric field)</td>
<td>$F = k_0 \frac{Q_1 Q_2}{r^2}$, $k_0 = 9 \times 10^9 (N \cdot m^2/C^2)$</td>
</tr>
<tr>
<td>Electric field</td>
<td>$E = \frac{V}{d}$, $F = qE$, $W = qV$</td>
</tr>
<tr>
<td>Electrical capacity</td>
<td>$Q = CV$, $C = \varepsilon \varepsilon_0 \frac{S}{d}$</td>
</tr>
<tr>
<td>Electrical capacity (parallel, series)</td>
<td>$C = C_1 + C_2$, $\frac{1}{C} = \frac{1}{C_1} + \frac{1}{C_2}$</td>
</tr>
<tr>
<td>Dielectric constant $\varepsilon$ (Relative dielectric constant $\varepsilon$)</td>
<td>$D = \varepsilon_0 E$, $C = \varepsilon_0 C_0$</td>
</tr>
<tr>
<td>Electrostatic energy</td>
<td>$U = \frac{1}{2} QV = \frac{1}{2} CV^2$</td>
</tr>
<tr>
<td>Electron in electrical field</td>
<td>$a = \frac{qE}{m}$, $\frac{1}{2} mv^2 = qV$</td>
</tr>
<tr>
<td>Coulomb's law (magnetic field)</td>
<td>$F = k_0 \frac{m_1 m_2}{r^2}$, $k_0 = \frac{10^7}{(4\pi)^2}$</td>
</tr>
<tr>
<td>Magnetic field $H$</td>
<td>$H = \frac{1}{2\pi r}$, $H = \frac{1}{2\pi r}$, $H = n I$</td>
</tr>
<tr>
<td>Magnetic field</td>
<td>$F = \mu_0 H</td>
</tr>
<tr>
<td>Magnetic flux density</td>
<td>$B = \frac{m}{4\pi r^2} = \mu_0 H$</td>
</tr>
<tr>
<td>Lorentz force</td>
<td>$F = Q \varepsilon B$, $r = \frac{mv}{QB}$</td>
</tr>
<tr>
<td>Electron in magnetic field</td>
<td>$\frac{1}{2} m v^2 = Q^2 B^2 r^2$, $\omega = \frac{v}{r} = \frac{QB}{m}$</td>
</tr>
<tr>
<td>Faraday's law of induction</td>
<td>$V = -n \frac{d \phi}{dt}$</td>
</tr>
<tr>
<td>Electromagnetic induction</td>
<td>$V = E \ell = vB \ell$, $I = \frac{vB \ell}{R}$</td>
</tr>
<tr>
<td>Mutual induction</td>
<td>$V_2 = -M \frac{d I_1}{dt}$</td>
</tr>
<tr>
<td>Self-induction</td>
<td>$V' = -L \frac{d I}{dt}$</td>
</tr>
</tbody>
</table>

**OPERATION**

5936 (UB) **Coulomb's law (electric f.)** $F = k_0 \frac{Q_1 Q_2}{r^2}$, $k_0 = 9 \times 10^9 (N \cdot m^2/C^2)$

( or ) scrolls to the following formula, ( ) to the previous formula, ( ) to the first formula, and ( ) to the last (17th) formula.
**EXAMPLE**

Display a desired scientific (electric and magnetic field) formula.

<table>
<thead>
<tr>
<th>Formula</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{Electric field} )</td>
<td>( E = \frac{V}{d} ) ( F = qE ) ( W = qV ) (Formula 2)</td>
</tr>
<tr>
<td>( \text{Electrons in electric field} )</td>
<td>( a = \frac{qE}{m} ) ( \frac{1}{2} m v^2 = qV ) (Formula 7)</td>
</tr>
<tr>
<td>( \text{Self-induction} )</td>
<td>( V = L \frac{di}{dt} ) (Formula 17)</td>
</tr>
<tr>
<td>( \text{Electrons in magnetic field} )</td>
<td>( \frac{1}{2} m v^2 = qE_{\text{field}}/\mu_0 ) ( w^2 v/r = \Omega B/m ) (Formula 13)</td>
</tr>
<tr>
<td>( \text{Lorentz force} )</td>
<td>( F = qvB ) ( F = qvB ) (Formula 12)</td>
</tr>
<tr>
<td>( \text{Coulomb's law in electric field} )</td>
<td>( F = k_0 \frac{Q_1 Q_2}{r^2} ) ( k = 8 \times 10^{-19} \text{N} \cdot \text{m}^2/\text{C}^2 ) (Formula 1)</td>
</tr>
</tbody>
</table>
Displays the following 13 scientific formulas:

<table>
<thead>
<tr>
<th>NAME</th>
<th>FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute temperature</td>
<td>( T(\text{K}) = t(\text{°C}) + 273.15 )</td>
</tr>
<tr>
<td>Heat capacity</td>
<td>( Q = CT = mcT )</td>
</tr>
<tr>
<td>Mechanical equivalent of heat</td>
<td>( W = JQ, ; J = 4.19 \text{(J/cal)} )</td>
</tr>
<tr>
<td>Boyle's law</td>
<td>( PV = \text{Constant} ) ( T = \text{constant} )</td>
</tr>
<tr>
<td>Thermal expansion (volume and temperature)</td>
<td>( V = V_0 (1 + \frac{T}{273}) )</td>
</tr>
<tr>
<td>Charles' law</td>
<td>( \frac{V}{V_0} = \frac{T}{T_0} )</td>
</tr>
<tr>
<td>Equation of state</td>
<td>( PV = nRT, ; R = 8.31 \text{(J/°K)} )</td>
</tr>
<tr>
<td>Law of partial pressures</td>
<td>( P = P_1 + P_2 + P_3 + \cdots )</td>
</tr>
<tr>
<td>Pressure</td>
<td>( P = \frac{1}{3}nmt^2 )</td>
</tr>
<tr>
<td>Internal energy</td>
<td>( U = \frac{1}{2}nm^2 ) ( N = \frac{3}{2}nRT )</td>
</tr>
<tr>
<td>Specific heat</td>
<td>( C_v = \frac{4U}{4T} - \frac{3R}{2}, ; C_p = \frac{4U}{4T} + \frac{5R}{2} )</td>
</tr>
<tr>
<td>Half life</td>
<td>( N = N_0 \left( \frac{1}{2} \right)^x ) ( (x = \frac{1}{T}) )</td>
</tr>
<tr>
<td>Mass-energy relation</td>
<td>( E = mc^2 )</td>
</tr>
</tbody>
</table>

**OPERATION**

5938 \[ UB \] **Absolute temperature** \( T(\text{K}) = t(\text{°C}) + 273.15 \) \[ (1) \]

\[ \text{scrolls to the next formula, to the previous formula, } \text{to the first formula, } \text{and } \text{to the last (13th) formula.} \]

**EXAMPLE**

Display a desired scientific (thermodynamic and others) formula.

[Formula 1] (Formula 2)
[Formula 6] (Formula 13)
[Formula 7] (Formula 1)
Displays the following 30 conversion formulas. Pressing  \[ \text{EQ} \] stores the currently displayed formula which then can be applied for calculation.

<table>
<thead>
<tr>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>(cm)</td>
<td>\times 0.01 (m)</td>
<td>(ft)</td>
<td>\times 30.48 (cm)</td>
</tr>
<tr>
<td></td>
<td>0.393701 (in)</td>
<td></td>
<td>0.3048 (m)</td>
</tr>
<tr>
<td></td>
<td>0.0328084 (ft)</td>
<td></td>
<td>12 (in)</td>
</tr>
<tr>
<td></td>
<td>0.0109361 (yd)</td>
<td></td>
<td>0.333333 (yd)</td>
</tr>
<tr>
<td></td>
<td>0.00000621371 (mile)</td>
<td></td>
<td>0.000189394 (mile)</td>
</tr>
<tr>
<td>(m)</td>
<td>\times 100 (cm)</td>
<td>(yd)</td>
<td>\times 91.44 (cm)</td>
</tr>
<tr>
<td></td>
<td>39.3701 (in)</td>
<td></td>
<td>0.9144 (m)</td>
</tr>
<tr>
<td></td>
<td>3.28084 (ft)</td>
<td></td>
<td>36 (in)</td>
</tr>
<tr>
<td></td>
<td>1.09361 (yd)</td>
<td></td>
<td>3 (ft)</td>
</tr>
<tr>
<td></td>
<td>0.00000621371 (mile)</td>
<td></td>
<td>0.000568182 (mile)</td>
</tr>
<tr>
<td>(in)</td>
<td>\times 2.54 (cm)</td>
<td>(mile)</td>
<td>\times 160934.4 (cm)</td>
</tr>
<tr>
<td></td>
<td>0.0254 (m)</td>
<td></td>
<td>1609.344 (m)</td>
</tr>
<tr>
<td></td>
<td>0.0633333 (ft)</td>
<td></td>
<td>63360 (in)</td>
</tr>
<tr>
<td></td>
<td>0.0277778 (yd)</td>
<td></td>
<td>5280 (ft)</td>
</tr>
<tr>
<td></td>
<td>0.0000157828 (mile)</td>
<td></td>
<td>1760 (yd)</td>
</tr>
</tbody>
</table>

**OPERATION**

**5950**

\[
\text{Metric conversion (length) (1)} \\
\times (cm) = 0.01 \times (m)
\]

\[ \text{EQ} \] scrolls to the following formula, \[ \text{EQ} \] to the previous formula, \[ \text{EQ} \] to the first formula, and \[ \text{EQ} \] to the last (30th) formula.

Pressing of \[ \text{EQ} \] executes a conversion of the currently displayed units.

**EXAMPLE**

Display a desired conversion formula.

\[ \text{EQ} \]  
\[
\text{Metric conversion (length) (2)} \\
\times (cm) = 0.393701 \times (in)
\]

\[ \text{EQ} \]  
\[
\text{Metric conversion (length) (5)} \\
\times (cm) = 0.00000621371 \times (mile)
\]

\[ \text{EQ} \]  
\[
\text{Metric conversion (length) (30)} \\
\times (mile) = 1760 \times (yd)
\]

\[ \text{EQ} \]  
\[
\text{Metric conversion (length) (26)} \\
\times (mile) = 160934.4 \times (cm)
\]

\[ \text{EQ} \]  
\[
\text{Metric conversion (length) (1)} \\
\times (cm) = 0.01 \times (m)
\]
EXAMPLE
Convert 110m and 300m to yards.

Metric conversion (length) [8]

\[
x(m) \rightarrow 1.09361x(yd)
\]

(Stores Formula 9 in memory)

110 m

\[
x(m)?110
\]
\[
x(yd)=120.2671
\]

(110m = 120.2671 yards)

300 m

\[
x(m)?300
\]
\[
x(yd)=328.083
\]

(300m = 328.083 yards)

* Once calculation is complete, a different conversion can be selected by first pressing \(=\) followed by the \(\leftarrow\) key.

IMPORTANT
This library function is executed by first storing the conversion formula into the formula storage memory. Note that the current formula memory contents are cleared by this procedure.
Displays the following 12 conversion formulas. Pressing the stores the currently displayed formula which then can be applied for calculation.

### Table: Metric Conversions for Area

<table>
<thead>
<tr>
<th>Conversion Unit</th>
<th>Conversion Formula</th>
<th>Conversion Unit</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>m²</td>
<td>x</td>
<td>0.01</td>
<td>a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.000247105</td>
<td>acre</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.000000386102</td>
<td>mile²</td>
</tr>
<tr>
<td>a</td>
<td></td>
<td>100</td>
<td>m²</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.0247105</td>
<td>acre</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.0000386102</td>
<td>mile²</td>
</tr>
</tbody>
</table>

#### Operation

**5960**

1. Press **Metric conversion (area)** once to display the following formula.

2. **Scroll** to the following formula, **↑** to the previous formula, **↓** to the first formula, and **←** to the last (12th) formula.

3. **Pressing** executes a conversion of the currently displayed units.

#### Example

Display a desired conversion formula.

<table>
<thead>
<tr>
<th><strong>5960</strong></th>
<th><strong>Metric conversion (area)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formula 2</strong></td>
<td><strong>x (m²) = 0.000247105 X (acre)</strong></td>
</tr>
<tr>
<td><strong>Formula 5</strong></td>
<td><strong>x (m²) = 0.000000386102 X (mile²)</strong></td>
</tr>
<tr>
<td><strong>Formula 12</strong></td>
<td><strong>x (mile²) = 640 X (acre)</strong></td>
</tr>
<tr>
<td><strong>Formula 10</strong></td>
<td><strong>x (mile²) = 2389990 X (m²)</strong></td>
</tr>
<tr>
<td><strong>Formula 1</strong></td>
<td><strong>x (m²) = 0.01 X (a)</strong></td>
</tr>
</tbody>
</table>

#### Sample

Convert 300m² to acres.

<table>
<thead>
<tr>
<th><strong>5960</strong></th>
<th><strong>Metric conversion (area)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formula 2</strong></td>
<td><strong>x (m²) = 0.000247105 X (acre)</strong></td>
</tr>
<tr>
<td><strong>EXE</strong></td>
<td><strong>x (m²) = 760</strong></td>
</tr>
<tr>
<td><strong>300 2N</strong></td>
<td><strong>x (acre) = 0.0741315</strong></td>
</tr>
</tbody>
</table>

Once calculation is complete, a different conversion can be selected by first pressing followed by the **EXE** key.
IMPORTANT
This library function is executed by first storing the conversion formula into the formula storage memory. Note that the current formula memory contents are cleared by this procedure.

METRIC CONVERSIONS FOR VOLUME

Displays the following 30 conversion formulas. Pressing \( \text{5970} \) stores the currently displayed formula which then can be applied for calculation.

<table>
<thead>
<tr>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x )</td>
<td>( \times ) ( 0.000001 ) ( [\text{m}^3] )</td>
<td>( \text{ft}^3 )</td>
<td>( 26316.8 ) ( [\text{cm}^3] )</td>
</tr>
<tr>
<td></td>
<td>0.0610237 ( [\text{in}^3] )</td>
<td></td>
<td>0.0283168 ( [\text{m}^3] )</td>
</tr>
<tr>
<td></td>
<td>0.0000353147 ( [\text{ft}^3] )</td>
<td>1728 ( [\text{in}^3] )</td>
<td>28.3168 ( [\text{ft}^3] )</td>
</tr>
<tr>
<td></td>
<td>0.001 ( [\text{l}] )</td>
<td></td>
<td>7.48052 ( [\text{gal(US)}] )</td>
</tr>
<tr>
<td></td>
<td>0.000264172 ( [\text{gal(US)}] )</td>
<td></td>
<td>6.22882 ( [\text{gal(UK)}] )</td>
</tr>
<tr>
<td></td>
<td>0.000219968 ( [\text{gal(UK)}] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \text{m}^3 )</td>
<td>1000000 ( [\text{cm}^3] )</td>
<td>( \times ) 1</td>
<td>( \times ) 1000 ( [\text{cm}^3] )</td>
</tr>
<tr>
<td></td>
<td>61023.7 ( [\text{in}^3] )</td>
<td></td>
<td>0.001 ( [\text{m}^3] )</td>
</tr>
<tr>
<td></td>
<td>35.3147 ( [\text{ft}^3] )</td>
<td></td>
<td>61.0237 ( [\text{in}^3] )</td>
</tr>
<tr>
<td></td>
<td>1000 ( [\text{l}] )</td>
<td></td>
<td>0.0353147 ( [\text{ft}^3] )</td>
</tr>
<tr>
<td></td>
<td>264.172 ( [\text{gal(US)}] )</td>
<td></td>
<td>0.264172 ( [\text{gal(US)}] )</td>
</tr>
<tr>
<td></td>
<td>219.968 ( [\text{gal(UK)}] )</td>
<td></td>
<td>0.219966 ( [\text{gal(UK)}] )</td>
</tr>
<tr>
<td>( \text{in}^3 )</td>
<td>16.3871 ( [\text{cm}^3] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0000163871 ( [\text{m}^3] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.000578704 ( [\text{ft}^3] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0163871 ( [\text{l}] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00432900 ( [\text{gal(US)}] )</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00360464 ( [\text{gal(UK)}] )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

OPERATION

5970 \( \text{LB} \) | Metric conversion (volume) \( \frac{[\text{cm}^3]}{[\text{m}^3]} \)

\( \text{LB} \) scrolls to the following formula, \( \text{LB} \) to the previous formula, \( \text{LB} \) to the first formula, and \( \text{LB} \) to the last (30th) formula.
Pressing of \( \text{LB} \) executes a conversion of the currently displayed units.
EXAMPLE

Display a desired conversion formula.

metrical conversion (volume) \[ \frac{\text{cm}^3}{\text{m}^3} = 0.000001 \] (Formula 1)

metrical conversion (volume) \[ \frac{\text{m}^3}{\text{in}^3} = 61023.71 \] (Formula 2)

metrical conversion (volume) \[ \frac{x}{\text{cm}^3} = 1000000x(\text{cm}^3) \] (Formula 7)

metrical conversion (volume) \[ \frac{x}{\text{gal(US)}} = 0.00454609x(\text{m}^3) \] (Formula 26)

metrical conversion (volume) \[ \frac{x}{\text{gal(UK)}} = 0.00264172x(\text{m}^3) \] (Stores Formula 5 in memory and executes)

metrical conversion (volume) \[ \frac{x}{\text{cm}^3} = 1500 \] (1800cm³ = approximately 0.48 gallons)

EXAMPLE

Convert 1800cm³ to gallons (US).

metrical conversion (volume) \[ \frac{x}{\text{cm}^3} = 1500 \] (Stores Formula 5 in memory and executes)

metrical conversion (volume) \[ \frac{x}{\text{cm}^3} = 1500 \] (1800cm³ = approximately 0.48 gallons)

Once calculation is complete, a different conversion can be selected by first pressing \( \text{M} \) followed by the \( \text{C} \) key.

IMPORTANT
This library function is executed by first storing the conversion formula into the formula storage memory. Note that the current formula memory contents are cleared by this procedure.
Displays the following 12 conversion formulas. Pressing \( \Box \) stores the currently displayed formula which then can be applied for calculation.

<table>
<thead>
<tr>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
<th>CONVERSION UNIT</th>
<th>CONVERSION FORMULA</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{g} )</td>
<td>( \times 0.001 \times \text{kg} )</td>
<td>( \text{oz} )</td>
<td>( \times 28.3495 \times \text{g} )</td>
</tr>
<tr>
<td></td>
<td>0.0352740 (oz)</td>
<td></td>
<td>0.0283495 (kg)</td>
</tr>
<tr>
<td></td>
<td>0.00220462 (lb)</td>
<td></td>
<td>0.0625 (lb)</td>
</tr>
<tr>
<td>( \text{kg} )</td>
<td>1000 ( \times \text{g} )</td>
<td>( \text{lb} )</td>
<td>453.59237 ( \times \text{g} )</td>
</tr>
<tr>
<td></td>
<td>35.2740 (oz)</td>
<td></td>
<td>0.45359237 (kg)</td>
</tr>
<tr>
<td></td>
<td>2.20462 (lb)</td>
<td></td>
<td>16 ( \times \text{oz} )</td>
</tr>
</tbody>
</table>

**OPERATION**

5980 \( \text{LB} \)

This displays the formula. \( \Box \) scrolls to the following formula, \( \Box \) to the previous formula, \( \Box \) to the first formula, and \( \Box \) to the last (12th) formula. Pressing of \( \Box \) executes a conversion of the currently displayed units.

**EXAMPLE**

Display a desired conversion formula.

\( \Box \)

\[ \text{Metric conversion (weight) } [1] \]

\[ \text{x (g)} = 0.001 \times \text{x (kg)} \]

\( \Box \) \( \Box \) \( \Box \)

\[ \text{Metric conversion (weight) } [2] \]

\[ \text{x (g)} = 0.0352740 \times \text{x (oz)} \]

(FORMULA 2)

\( \Box \)

\[ \text{Metric conversion (weight) } [6] \]

\[ \text{x (kg)} = 2.20462 \times \text{x (lb)} \]

(FORMULA 6)

\( \Box \)

\[ \text{Metric conversion (weight) } [12] \]

\[ \text{x (lb)} = 16 \times \text{x (oz)} \]

(FORMULA 12)

\( \Box \) \( \Box \)

\[ \text{Metric conversion (weight) } [9] \]

\[ \text{x (oz)} = 0.0625 \times \text{x (lb)} \]

(FORMULA 9)

\( \Box \)

\[ \text{Metric conversion (weight) } [1] \]

\[ \text{x (g)} = 0.001 \times \text{x (kg)} \]

(FORMULA 1)

**EXAMPLE**

Convert 2.5kg to ounces.

\( \Box \) \( \Box \) \( \Box \) \( \Box \)

\[ \text{Metric conversion (weight) } [5] \]

\[ \text{x (kg)} = 35.2740 \times \text{x (oz)} \]

(FORMULA 5)

2 \( \Box \) 5 \( \Box \)

\[ \text{x (kg)} = \frac{2.5}{35.2740} \times \text{x (oz)} = 0.07185 \text{ oz} \]

\( \Box \)

\[ \text{(Stores Formula 5 in memory and executes)} \]

\[ \text{(2.5kg = 88.185 ounces)} \]

* Once calculation is complete, a different conversion can be selected by first pressing \( \Box \) followed by the \( \Box \) key.

**IMPORTANT**

This library function is executed by first storing the conversion formula into the formula storage memory. Note that the current formula memory contents are cleared by this procedure.
Determines upper probability for normal distribution with five significant digits using the following formula:

\[ f(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx \]

**OPERATION**

\[ \text{U} \text{P} \text{E} \text{R} \text{ P} \text{R} \text{O} \text{B} \text{A} \text{L} \text{ Y} \text{ P} \text{R} \text{O} \text{B} \text{A} \text{L} \text{ I} \text{M} \text{E} \text{T} \text{R} \text{AL} \text{s} \text{ N}(0.1\#) \]

\[ x = 0 \] \[ 7 \]

**EXAMPLE**

Determine the upper probability for normal distribution when \( x = 1.53 \).

\[ \text{U} \text{P} \text{E} \text{R} \text{ P} \text{R} \text{O} \text{B} \text{A} \text{L} \text{ Y} \text{ P} \text{R} \text{O} \text{B} \text{A} \text{L} \text{ I} \text{M} \text{E} \text{T} \text{R} \text{AL} \text{s} \text{ N}(0.1\#) \]

\[ x = 1 \] \[ 53 \] \[ 7 \]

\[ d = 0.063008 \]

(Return to initial display)

Here, the upper probability integral is 0.063008.
Determines upper probability for \( x^2 \) distribution with five significant digits using the following formula:

\[
P(x^2, \nu) = \int_{x^2}^{\infty} \frac{1}{2\Gamma(\frac{\nu}{2})} \left(\frac{x^2}{2}\right)^{\frac{\nu}{2} - 1} e^{-x^2/2} dx^2 \quad (\nu: \text{degree of freedom})
\]

**OPERATION**

6220 UB

**EXAMPLE**

Determine the upper probability for \( x^2 \) distribution when degree of freedom (\( \nu \)) = 4, and \( x^2 = 2 \).

4 EXP

5 EXP

2 EXP

(The upper probability integral is 0.73576.)

6 EXP

(Return to initial display)
Determine upper probability for t distribution with five significant digits using the following formula:

\[
P(x, \nu) = \int_{x}^{\infty} \left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu + 1}{2}} \frac{1}{\sqrt{\nu} B\left(\frac{1}{2}, \frac{\nu}{2}\right)} \, dx
\]

\(x: \) degree of freedom

**Operation**

6230 UB

Upper probability \(t(x, \nu)\)

**Example**

Determine the upper probability for t distribution when degree of freedom \(\nu = 2\), and \(x = 2.92\).

2 0 92 0

Upper probability \(t(x, \nu)\)

(Degree of freedom)

Upper probability \(t(x, \nu)\)

(Enter value of \(x\))

Upper probability \(t(x, \nu)\)

(The upper probability integral is 0.05.)

Upper probability \(t(x, \nu)\)

(Return to initial display)
Determine upper probability for F distribution with five significant digits using the following formula:

\[ P(x, \nu_1, \nu_2) = \int_{x}^{\infty} \frac{x^{\frac{\nu_1}{2} - 1} \cdot x^{\frac{\nu_2}{2} - 1}}{B(\frac{\nu_1}{2}, \frac{\nu_2}{2}) (\nu_2 + \nu_1 x)^{\frac{\nu_2}{2}}} \, dx \quad (\nu_1: \text{degree of freedom 1}; \nu_2: \text{degree of freedom 2}) \]

**EXAMPLE**

Determine the upper probability for F distribution when degree of freedom 1 \((\nu_1) = 5\), degree of freedom 2 \((\nu_2) = 3\) and \(x = 9.01\).

5 \text{ KEE}\n
3 \text{ KEE}\n
9 + 01 \text{ KEE}\n
\text{Upper probability F(x, \nu_1, \nu_2) } \quad \text{(Degree of freedom 1)}\n
\text{Upper probability F(x, \nu_1, \nu_2) } \quad \text{(Degree of freedom 2)}\n
\text{Upper probability F(x, \nu_1, \nu_2) } \quad \text{(Value of x)}\n
\text{Upper probability F(x, \nu_1, \nu_2) } \quad \text{(The upper probability integral is 0.050026.)}\n
\text{Upper probability F(x, \nu_1, \nu_2) } \quad \text{(Return to initial display)}
Determine upper cumulative frequency for binomial distribution with five significant digits using the following formula:

\[ B(x, n, p) = \sum_{y=x}^{n} \binom{n}{y} p^y (1-p)^{n-y} \]

- **n**: maximum value of \( x \)
- **p**: probability
- **Q**: sum of frequencies produces past \( x \)

### Operation

**6310 UB**

**Cumulative frequency** \( B(x, n, p) \)

### Example

Determine the upper cumulative frequency for binomial distribution when the maximum value of \( x \) (\( n \)) = 5, probability (\( p \)) = 0.5 and \( x = 4 \).

<table>
<thead>
<tr>
<th>( x )</th>
<th><strong>Cumulative frequency</strong> ( B(x, n, p) )</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>( B(x, n, p) ) = 0.1875</td>
<td>(Maximum value of ( x ))</td>
</tr>
<tr>
<td>4</td>
<td>( B(x, n, p) ) = 0.1875</td>
<td>(Probability)</td>
</tr>
<tr>
<td>3</td>
<td>( B(x, n, p) )</td>
<td>(Value of ( x ))</td>
</tr>
<tr>
<td>2</td>
<td>( B(x, n, p) )</td>
<td>(The upper cumulative frequency is 0.1875.)</td>
</tr>
<tr>
<td>1</td>
<td>( B(x, n, p) )</td>
<td>(Return to initial display)</td>
</tr>
<tr>
<td>0</td>
<td>( B(x, n, p) )</td>
<td></td>
</tr>
</tbody>
</table>

287
Determine the upper cumulative frequency for Poisson distribution when mean value ($\lambda$) = 2, and $x = 4$. 

2 EXC

Cumulative frequency $P(x, \lambda)$

1. $x = 0$?

4 EXC

Cumulative frequency $P(x, \lambda)$

1. $x = 0$?

Cumulative frequency $P(x, \lambda)$

1. $x = 0$ 1.4228

EXC

Cumulative frequency $P(x, \lambda)$

1. $x = 2$?

(Mean value)

(Value of $x$)

(The upper cumulative frequency is 0.14228.)

(Return to initial display)
UPPER CUMULATIVE FREQUENCY (HYPERGEOMETRIC DISTRIBUTION)

Determines upper cumulative frequency for hypergeometric distribution with five significant digits using the following formula:

\[ H(x, n, M, N) = \sum_{i=0}^{x} \frac{\binom{M}{i} \binom{N-M}{n-i}}{\binom{N}{n}} \]

\[ P \text{: Sum of frequencies produces past } x \text{ (cumulative frequency)} \]

**OPERATION**

6330 ULR

**EXAMPLE**

Determine the upper cumulative frequency for hypergeometric distribution when \( N = 3 \), \( M = 2 \), \( n = 1 \), and \( x = 1 \).

<table>
<thead>
<tr>
<th>3 EX</th>
<th><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</th>
<th>(Value of N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 EX</td>
<td><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</td>
<td>(Value of M)</td>
</tr>
<tr>
<td>1 EX</td>
<td><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</td>
<td>(Value of n)</td>
</tr>
<tr>
<td>1 EX</td>
<td><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</td>
<td>(Value of x)</td>
</tr>
<tr>
<td>1 EX</td>
<td><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</td>
<td>(The upper cumulative frequency is 0.66667.)</td>
</tr>
<tr>
<td>1 EX</td>
<td><strong>Cumulative frequency</strong> ( H(x, n, M, N) )</td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

N = 3 ?
PERCENTAGE POINT NORMAL DISTRIBUTION

Determines percentage point for normal distribution with five significant digits using the following formula:

\[ x(p) : \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx = p \quad (p: \text{probability}) \]

**OPERATION**

6410 UB

Percentage points \( \mathcal{N}(\theta, \sigma) \)

**EXAMPLE**

Determine the percentage point for normal distribution when \( p = 0.05 \).

\[ 0 \leq 0.05 \]

Percentage points \( \mathcal{N}(\theta, \sigma) \)

\[ x = 1.6449 \]

(Probability)

Percentage points \( \mathcal{N}(\theta, \sigma) \)

\[ p = 0.05 \]

(The percentage point is 1.6449.)

(Probability)

(Probability)

(Return to initial display)
Determines percentage point for $x^2$ distribution with five significant digits using the following formula:

$$

F^{-1}(p) = \int_0^\infty \frac{1}{2^{\nu/2} \Gamma(\nu/2)} \left( \frac{x}{2} \right)^{\nu/2-1} e^{-x/2} dx = p

$$

($\nu$: degree of freedom, $\Gamma$: gamma function, $p$: probability)

**Example**

Determine the percentage point for $x^2$ distribution when degree of freedom ($\nu$) = 2, and probability $p = 0.5$.

<table>
<thead>
<tr>
<th>$\nu$</th>
<th>Percentage points $x^2(\nu, \nu)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Percentage points $x^2(1, \nu)$</td>
</tr>
<tr>
<td>2</td>
<td>Percentage points $x^2(2, \nu)$</td>
</tr>
<tr>
<td>3</td>
<td>Percentage points $x^2(3, \nu)$</td>
</tr>
</tbody>
</table>

(Degree of freedom)

<table>
<thead>
<tr>
<th>$p$</th>
<th>Percentage points $x^2(\nu, p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>Percentage points $x^2(2, 0.5)$</td>
</tr>
</tbody>
</table>

(Probability)

<table>
<thead>
<tr>
<th>$x^2$</th>
<th>Percentage points $x^2(\nu, p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3863</td>
<td>Percentage points $x^2(2, 0.5)$</td>
</tr>
</tbody>
</table>

(The percentage point is 1.3863)

<table>
<thead>
<tr>
<th>$\nu$</th>
<th>Percentage points $x^2(\nu, p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Percentage points $x^2(2, 0.5)$</td>
</tr>
</tbody>
</table>

(Return to initial display)
Determine percentage point for t distribution with five significant digits using the following formula:

\[ tp(\nu) = \int_{-\infty}^{\infty} \frac{t^2 - (\nu + 1)/2}{\nu B(\frac{1}{2}, \frac{\nu}{2})} dt = p \]

\( \nu \) : degree of freedom
\( p \) : probability

**OPERATION**

6430 UB

**EXAMPLE**

Determine the percentage point for t distribution when degree of freedom (\( \nu \)) = 1, and probability (\( p \)) = 0.05.

1 \( \Rightarrow \) Percentage points t(x, \( \nu \))

0 \( \Rightarrow \) 05 \( \Rightarrow \) Percentage points t(x, \( \nu \))

(\( \nu \) = 1)

(\( \nu \) = 6.3137)

(\( \nu \) = 0)

(Return to initial display)
Determines percentage point for F distribution with five significant digits using the following formula:

\[ F_p(v_1, v_2) : \int_{0}^{\infty} \frac{v_1 \frac{v_2}{v_2 + \frac{v_1}{F} - 1}}{\Gamma(v_2/2) \Gamma(v_1/2)} v_1^{v_1/2} v_2^{v_2/2} dF = p \]

\( v_1 : \) degree of freedom 1

\( v_2 : \) degree of freedom 2

\( p : \) probability

**OPERATION**

6440 [B]  

**EXAMPLE**

Determine the percentage point for F distribution when degree of freedom 1 \((v_1) = 2\), degree of freedom 2 \((v_2) = 3\) and probability \((p) = 0.05\).

2 [B]  
3 [B]  
0 : 05 [A]  

<table>
<thead>
<tr>
<th>Percentage points ( F(2, 3, 9) )</th>
<th>(Degree of freedom 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage points ( F(2, 3, 9) )</td>
<td>(Degree of freedom 2)</td>
</tr>
<tr>
<td>( p = 0.05 )</td>
<td>(Probability)</td>
</tr>
<tr>
<td>Percentage points ( F(2, 3, 9) )</td>
<td>(The percentage point is 9.5521)</td>
</tr>
<tr>
<td>( F = 9.5521 )</td>
<td>(Return to initial display)</td>
</tr>
<tr>
<td>Percentage points ( F(2, 3, 9) )</td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>
NORMA L RANDOM NUMBERS

Generates random numbers contained in the standard normal distribution N (0, 1²). This unit creates two independent normal random numbers (u, v) based upon two uniform random numbers (x, y).

\[ u = \sqrt{-2 \cdot \log x} \cdot \cos(2\pi y) \]
\[ v = \sqrt{-2 \cdot \log x} \cdot \sin(2\pi y) \]

OPERATION

6450 [LB]

EXAMPLE

Generate a series of normal random numbers.

\[ x_1 \]

\[ 0.6103300096 \]
\[ 0.5713331854 \]

\[ x_2 \]

\[ 0.5713331554 \]
\[ -1.6543404886 \]

\[ x_3 \]

\[ -1.6543064968 \]
\[ 0.5420575429 \]
EXPOENTIAL RANDOM NUMBERS

Generates random numbers contained in the exponential distribution \( E(\lambda, t) \). This unit creates random numbers in accordance with exponential distribution using uniform random numbers.

\[
1 = \frac{1}{\lambda} \ln x
\]

\[\lambda : \text{mean value}\]

**Operation**

\begin{align*}
60 \ & \text{UB} \\
E(1, t) \\
1 & = 1 \ ?
\end{align*}

**Sample**

Generate a series of exponential random numbers when the mean value \((\lambda) = 3\).

\[
E(1, t) \\
1 & = 1 \ ?
\]

\[
\begin{array}{|c|}
\hline
E(1, t) \\
0.1164873591 \\
0.1164873591 \\
0.7274505817 \\
0.7274505817 \\
0.6262426639 \\
\hline
\end{array}
\]

*Mean value*

*\(E(1, t)\)*

*\(1 = 1 \ ?\)*

Return to the mean value input display, first press \(\text{UB}\) to terminate the library. Next, press \(\text{UB}\) to return to the initial display.
SINGLE VARIABLE STATISTICS

Determines the following statistics and determines the deviation value for input of n data items.

<table>
<thead>
<tr>
<th>Number of data items</th>
<th>CNT : n</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum of data</td>
<td>SUMX : (\sum x)</td>
</tr>
<tr>
<td>Sum of squares of data</td>
<td>SUMX2 : (\sum x^2)</td>
</tr>
<tr>
<td>Mean of data</td>
<td>MEANX : (\frac{\sum x}{n})</td>
</tr>
<tr>
<td>Population standard deviation of data</td>
<td>SDXN : (x\sigma)</td>
</tr>
<tr>
<td>Sample standard deviation of data</td>
<td>SDX : (x\sigma - 1)</td>
</tr>
</tbody>
</table>

\[ \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n}} \]

\[ \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n(n-1)}} \]

OPERATION

6500 UB

The menu illustrated above is displayed for single variable statistical calculations. The following six items can be selected from this menu:

1. I : Data input (does not clear data already present in memory)
2. D : Data deletion (deletes erroneous or unnecessary data)
3. C : Data clear
4. L : Statistic display
   Displays number of data items, sum of data, sum of squares of data, mean of data, population standard deviation of data, and sample standard deviation of data in sequence. (or ) scrolls to the following data item, (or ) to the previous data item, and (or ) terminate statistic display.
5. T : Calculates deviation value of obtained value.
6. P : Outputs all statistics to printer

EXAMPLE

Enter the following five test scores and display statistics. Also determine the deviation value for the score of 88.

Data : 98, 88, 62, 90, 78

C (Data clear)

Y (Data clear confirmation)

I (Data input)
Here, the deviation value of the 88 score is 53.9.
SINGLE VARIABLE STATISTICS FLOWCHART

Library start

1. Data input
   - Data input
   - X only? YES
     - NO

2. Data deletion
   - Data input to be deleted
   - X only? YES
     - NO

3. Data clear
   - Process selection
   - Data clear

4. Statistic display
   - First statistic set
   - Statistic display
     - X
     - Next statistic display
     - Previous statistic display
     - Statistical display
     - YES
     - NO

5. Deviation value calculation
   - y input
   - X only? YES
     - NO
     - Deviation value display

6. Printer output
   - Statistic output to printer
Performs linear regression analysis on n data groups (x, y) and calculates the statistics listed below. Also determines the following on the regression line:

Estimated value of x in relation to y (EOX)
Estimated value of y in relation to x (EOY)

**TATISTIC TABLE**

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Formula/Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of data items</td>
<td>CNT : n</td>
</tr>
<tr>
<td>Sum of x data</td>
<td>SUMX : Σx</td>
</tr>
<tr>
<td>Sum of y data</td>
<td>SUMY : Σy</td>
</tr>
<tr>
<td>Sum of squares of x data</td>
<td>SUMX² : Σx²</td>
</tr>
<tr>
<td>Sum of squares of y data</td>
<td>SUMY² : Σy²</td>
</tr>
<tr>
<td>Sum of products of x and y data</td>
<td>SUMXY : Σxy</td>
</tr>
<tr>
<td>Mean of x data</td>
<td>MEANX : Σx/n</td>
</tr>
<tr>
<td>Mean of y data</td>
<td>MEANY : Σy/n</td>
</tr>
<tr>
<td>Population standard deviation of x data</td>
<td>SDXN : xσn  = √(Σx²/n - (Σx)²/n²)</td>
</tr>
<tr>
<td>Population standard deviation of y data</td>
<td>SDYN : yσn  = √(Σy²/n - (Σy)²/n²)</td>
</tr>
<tr>
<td>Sample standard deviation of x data</td>
<td>SDX : xσn-1 = √((n-1)/(n-1)) Σx² - (Σx)²/n(n-1)</td>
</tr>
<tr>
<td>Sample standard deviation of y data</td>
<td>SDY : yσn-1 = √((n-1)/(n-1)) Σy² - (Σy)²/n(n-1)</td>
</tr>
<tr>
<td>Regression constant term</td>
<td>LRA : a  = (Σxy - Σx Σy)/n</td>
</tr>
<tr>
<td>Regression coefficient</td>
<td>LRB : b  = (nΣx² - (Σx)²)(nΣy² - (Σy)²)</td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>COR : r  = √((nΣx² - (Σx)²)(nΣy² - (Σy)²))</td>
</tr>
</tbody>
</table>

**ERATION**

10 ubs

The menu illustrated above is displayed for linear regression calculations. The following seven options can be selected from this menu:

I: Data input
D: Data deletion (deletes erroneous or unnecessary data)
C: Data clear
4. **L**: Statistic display
Displays number of data items, sum of x data, sum of y data, sum of squares of x data, sum of squares of y data, sum of products of x and y data, mean of x data, mean of y data, population standard deviation of x data, population standard deviation of y data, sample standard deviation of x data, sample standard deviation of y data, linear regression constant term, linear regression coefficient, and correlation coefficient. **C** (or **D**) scrolls to the following data item, **E** to the previous data item, and **F** or **G** terminates statistic display.

5. **X**: Calculates x value for y on regression line.
6. **Y**: Calculates y value for x on regression line.
7. **P**: Outputs all statistics to printer.

**EXAMPLE**
Enter the following five sets of height/weight, and display statistics. Also estimate the weight for a person whose height is 170cm.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Height (x)</strong></td>
<td>160</td>
<td>158</td>
<td>175</td>
<td>163</td>
<td>172</td>
</tr>
<tr>
<td><strong>Weight (y)</strong></td>
<td>43</td>
<td>45</td>
<td>60</td>
<td>46</td>
<td>58</td>
</tr>
</tbody>
</table>

**C**
Reggression analysis \(y = a + bx\)
(Data clear)

**Y**
Reggression analysis \(y = a + bx\)
(Data clear confirmation)

**1**
Input data \((x, y)\)
(Data input)

**160** **EX**
Input data \((x, y)\)
(x input)

**43** **EX**
Input data \((x, y)\)
(y input)

**158** **EX** **45** **EX** **175**
Input data \((x, y)\)
(Remaining x, y data input)

**EX**
Reggression analysis \(y = a + bx\)
(Return to menu display)

**L**
CNT : n = 5
SUMX : Ix = 828
(Sum of sum of data and sum of x data)

SUMX : Ix = 828
SUMY : Iy = 252
(Sum of y data)

SUMX2 : Ix\(^2\) = 137348
SUMY2 : Iy\(^2\) = 12854
(Sum of squares of x data)

SUMX2 : Ix\(^2\) = 137348
SUMY2 : Iy\(^2\) = 12854
(Sum of squares of y data)

SUMXY : Ix \cdot Iy = 41884
(Sum of products of x and y data)

MEANX : Ix/n = 165.6
(Mean of x data)

MEANX : Iy/n = 50.4
(Mean of y data)

MEANX : Iy/n = 50.4
(Mean of y data)

MEANX : Ix/n = 165.6
(Sum of products of x and y data)

SDXN : x\(\cdot\)n = 6.71114694
(Population standard deviation of x data)
<table>
<thead>
<tr>
<th>SDXN</th>
<th>$x_{\text{RN}} = 6.71184694$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDYN</td>
<td>$y_{\text{RN}} = 7.11617875$</td>
</tr>
<tr>
<td>SDX</td>
<td>$x_{\text{RN}-1} = 7.503332593$</td>
</tr>
<tr>
<td>SDY</td>
<td>$y_{\text{RN}-1} = 7.956129712$</td>
</tr>
</tbody>
</table>

**Population standard deviation of $y$ data**

**Sample standard deviation of $x$ data**

**Sample standard deviation of $y$ data**

**Linear regression constants (x)**

**Linear regression coefficient**

**Correlation coefficient**

**Regression analysis** \( y = a + bx \)

**Estimation of $y$** (weights)

**Estimated value for weight following input of height**

**Regression analysis** \( y = a + bx \)

**Estimation of $y$** (weights)

**Regression analysis** \( y = a + bx \)

**Return to menu display**

**Return to menu display**

These data produce the line \( y = -120.7886323 + 1.03374778x \). Also, input of a height 170cm results in an estimated weight of 54.9kg.
Performs logarithmic regression analysis on n data groups (x, y) and calculates the statistics listed below. Also determines the following on the logarithmic curve:

- Estimated value of x in relation to y (EOX)
- Estimated value of y in relation to x (EOY)

**STATISTIC TABLE**

<table>
<thead>
<tr>
<th>Number of data items</th>
<th>CNT ( : n )</th>
<th>SUMLNX ( : \Sigma \ln x )</th>
<th>( \Sigma (\ln x) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum of x data logarithmic values</td>
<td>SUMY ( : \Sigma y )</td>
<td>SUMLNX2 ( : \Sigma \ln x^2 )</td>
<td>( \Sigma (\ln x)^2 )</td>
</tr>
<tr>
<td>Sum of y data</td>
<td>SUMY2 ( : \Sigma y^2 )</td>
<td>SUMLNXXY ( : \Sigma \ln x \cdot y )</td>
<td>( \Sigma (\ln x) \cdot y )</td>
</tr>
<tr>
<td>Sum of squares of x data logarithmic values</td>
<td>MEANLNX ( : \Sigma \ln x / n )</td>
<td>( \Sigma (\ln x) / n )</td>
<td></td>
</tr>
<tr>
<td>Sum of squares of y data</td>
<td>MEANY ( : \Sigma y / n )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum of products of x data logarithmic values and of y data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean of x data logarithmic values</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean of y data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Population standard deviation of x data logarithmic values</td>
<td>SDLINXN ( : \ln x_{\text{an}} )</td>
<td>( \sqrt{n \Sigma (\ln x)^2 - (\Sigma \ln x)^2} / n )</td>
<td></td>
</tr>
<tr>
<td>Population standard deviation of y data</td>
<td>SDLINX ( : \ln x_{\text{an}-1} )</td>
<td>( \sqrt{n \Sigma y^2 - (\Sigma y)^2} / n )</td>
<td></td>
</tr>
<tr>
<td>Sample standard deviation of x data logarithmic values</td>
<td>SDYN ( : y_{\text{an}} )</td>
<td>( \sqrt{n \Sigma y^2 - (\Sigma y)^2} / (n(n-1)) )</td>
<td></td>
</tr>
<tr>
<td>Sample standard deviation of y data</td>
<td>SDY ( : y_{\text{an}-1} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regression constant term</td>
<td>RA ( : a )</td>
<td>( \frac{\sum \ln y - \sum \ln x \cdot \sum y}{n} )</td>
<td></td>
</tr>
<tr>
<td>Regression coefficient</td>
<td>RB ( : b )</td>
<td>( \frac{\sum \ln y - \sum \ln x \cdot \sum y}{n \Sigma (\ln x)^2 - (\Sigma \ln x)^2} )</td>
<td></td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>COR ( : r )</td>
<td>( \sqrt{n \Sigma (\ln x)^2 - (\Sigma \ln x)^2)(n \Sigma y^2 - (\Sigma y)^2)} )</td>
<td></td>
</tr>
</tbody>
</table>

**OPERATION**

The menu illustrated above is displayed for logarithmic regression calculations. The following seven items can be selected from this menu:

1. D: Data input
2. D: Data deletion (deletes erroneous or unnecessary data)
3. C: Data clear
L: Statistic display
Displays number of data items, sum of x data logarithmic values, sum of y data, sum of squares of x data logarithmic values, sum of squares of y data, sum of products of x data logarithmic values and y data, mean of x data logarithmic values, mean of y data, population standard deviation of x data logarithmic values, population standard deviation of y data, sample standard deviation of x data logarithmic values, sample standard deviation of y data, regression constant term, regression coefficient, and correlation coefficient. (X) (or EX) scrolls to the following data item, (C) to the previous data item, and (O) or (E) terminates statistic display.

X: Calculates x value for y on logarithm curve.
Y: Calculates y value for x on logarithm curve.
P: Outputs all statistics to printer.

**SAMPLE**

Enter the following measured data for microbes, perform logarithmic regression, and display the statistics. Also estimate the number of microbes with a temperature of 18 degrees using the logarithm curve obtained.

<table>
<thead>
<tr>
<th>Temperature (x)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microbes (y)</td>
<td>680</td>
<td>1100</td>
<td>1300</td>
<td>1440</td>
<td>1600</td>
</tr>
</tbody>
</table>

- **Regression analysis**
  - **Clear data** (Y/N)?
  - **Input data (x, y)**
    - 1100
  - **Input data (x, y)**
    - 1300
  - **Input data (x, y)**
    - 27
  - **Input data (x, y)**
    - 1440
  - **Input data (x, y)**
    - 36

(Data clear)
(Data clear confirmation)
(Data input)
(x input)
(y input)

Remaining x, y data input)
(Return to menu display)
(Statistic display showing number of data and sum of x data logarithmic values)
(Sum of y data)
(Sum of squares of x data logarithmic values)
(Sum of squares of y data)
(Sum of products of x data logarithmic values and y data)
(Mean of x data logarithmic values)
(Mean of y data)
Here, these data produce the curve $y = -52.62523046 + 456.935247 \cdot \ln x$. Also, input of a temperature of 18 degrees results in an estimated total of 1,268 microbes.
EXponential Regression Analysis

\( y = ab^x \)

performs exponential regression analysis on \( n \) data groups \((x, y)\) and calculates the statistics listed below. Also determines the following on the exponential curve:

Estimated value of \( x \) in relation to \( y \) (EOX)
Estimated value of \( y \) in relation to \( x \) (EOY)

**Statistical Table**

<table>
<thead>
<tr>
<th>Description</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of data items</td>
<td>( CNT : n )</td>
</tr>
<tr>
<td>Sum of ( x ) data</td>
<td>( SUMX : \Sigma x )</td>
</tr>
<tr>
<td>Sum of ( y ) data logarithmic values</td>
<td>( SUMLNY : \Sigma lny )</td>
</tr>
<tr>
<td>Sum of squares of ( x ) data</td>
<td>( SUMX2 : \Sigma x^2 )</td>
</tr>
<tr>
<td>Sum of squares of ( y ) data logarithmic values</td>
<td>( SUMLNY2 : \Sigma lny^2 )</td>
</tr>
<tr>
<td>Sum of products of ( x ) data and ( y ) data</td>
<td>( SUMXLNY : \Sigma xlny )</td>
</tr>
<tr>
<td>Mean of ( x ) data</td>
<td>( MEANX : \Sigma x/n )</td>
</tr>
<tr>
<td>Mean of ( y ) data logarithmic values</td>
<td>( MEANLNY : \Sigma lny/n )</td>
</tr>
<tr>
<td>Population standard deviation of ( x )</td>
<td>( SDXX : \sigma_x )</td>
</tr>
<tr>
<td>Population standard deviation of ( y )</td>
<td>( SDLNYX : \sigma_{lny} )</td>
</tr>
<tr>
<td>Sample standard deviation of ( x )</td>
<td>( SDX : \sigma_{x-1} )</td>
</tr>
<tr>
<td>Sample standard deviation of ( y )</td>
<td>( SDLNY : \sigma_{lny-1} )</td>
</tr>
<tr>
<td>Regression constant term</td>
<td>( RA : a )</td>
</tr>
<tr>
<td>Regression coefficient</td>
<td>( RB : b )</td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>( COR : r )</td>
</tr>
</tbody>
</table>

\( n(\Sigma x^2 - (\Sigma x)^2/n) \)

\( \sqrt{n(\Sigma (lny)^2 - (\Sigma lny)^2/n)} \)

\( n(\Sigma x - (\Sigma x/n))^2/n(n-1) \)

\( \sqrt{n(\Sigma lny^2 - (\Sigma lny)^2/n)} \)

\( \Sigma (lny - b \Sigma x) \)

\( \Sigma (n(\Sigma lny - \Sigma x \cdot \Sigma lny)/n^2 - \Sigma x^2/n^2) \)

\( \sqrt{(n(\Sigma x^2/n - (\Sigma x)^2/n^2))(n(\Sigma lny^2/n - (\Sigma lny)^2/n)^2)} \)

**Operation**

- **Menu**:
  - Regression analysis \((y = ab^x)\)
  - In, Del, Clear, List, eox, eoy, P, ?

The screen illustrated above is displayed for exponential regression calculations. The following seven items can be selected from this menu:

- **I**: Data input
- **D**: Data deletion (deletes erroneous or unnecessary data)
- **C**: Data clear
4. L: Statistic display
   Displays number of data items, sum of x data, sum of y data logarithmic values, sum of squares of x data, sum of squares of y data logarithmic values, sum of products of x data and y data logarithmic values, mean of x data, mean of y data logarithmic values, population standard deviation of x data, population standard deviation of y data logarithmic values, sample standard deviation of x data, sample standard deviation of y data logarithmic values, regression constant term, regression coefficient, and correlation coefficient. < or > scrolls to the following data item, and < or > terminates statistic display.

5. X: Calculates x value for y on regression line.
6. Y: Calculates y value for x on regression line.
7. P: Outputs all statistics to printer.

**EXAMPLE**

Enter the following data for the amount of sales per customer and number of customers for a store, perform exponential regression, and display the statistics. Also estimate the amount of sales per customer for 150 customers using the exponential curve obtained.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customers (x)</td>
<td>115</td>
<td>124</td>
<td>130</td>
<td>138</td>
<td>142</td>
</tr>
<tr>
<td>Sales/customer (y)</td>
<td>40</td>
<td>41.6</td>
<td>43.0</td>
<td>46.0</td>
<td>46.5</td>
</tr>
</tbody>
</table>

(Data clear)

(Data clear confirmation)

(Data input)

(x input)

(y input)

(Remaining x, y data input)

(Return to menu display)

(Statistic display showing number of data and sum of x data)

(Sum of y data logarithmic values)

(Sum of squares of x data)

(Sum of squares of y data logarithmic values)

(Sum of products of x data and y data logarithmic values)

(Mean of x data)
\[
\begin{align*}
\text{MEAN}_X & : \bar{x}/n = 129.8 \\
\text{MEAN}_Y & : \bar{y}/n = 3.769254689 \\
\text{SD}_X & : \sigma_X = 9.6297475 \\
\text{SD}_Y & : \sigma_Y = 5.774540647E-02 \\
\text{SD}_X \cdot \text{SD}_Y & : \rho_{xy} = 5.774540647E-02 \\
\text{SD}_X \cdot \text{SD}_Y : \rho_{xy} & = 5.774540647E-02 \\
\text{SD}_X & : \sigma_X = 10.82589488 \\
\text{SD}_Y & : \sigma_Y = 6.455244516E-02 \\
\text{SD}_X \cdot \text{SD}_Y & : \rho_{xy} = 6.455244516E-02 \\
\text{SD}_X & : \sigma_X = 6.455244516E-02 \\
\text{SD}_Y & : \sigma_Y = 20.1317721 \\
\text{RA} & : a = 1.005926239 \\
\text{RB} & : b = 0.9907846423 \\
\text{COR} & : r = 1.005926239 \\
\end{align*}
\]

**Mean of y data logarithmic values**

**Population standard deviation of x data**

**Population standard deviation of y data logarithmic values**

**Sample standard deviation of x data**

**Sample standard deviation of y data logarithmic values**

**Regression constant term**

**Regression coefficient**

**Correlation coefficient**

**Regression analysis \( y = ab^x \)**

**Estimation of y \( y = ab^x \)**

**Estimated value for y following input of 150 customers**

**Return to menu display**

- \( y = 20.1317721 \times 1.005926239 \). Also, input of a total 50 customers results in an estimated amount per customer of $48,843.
POWER REGRESSION ANALYSIS

\( y = ax^b \)

Performs power regression analysis on \( n \) data groups \((x, y)\) and calculates the statistics listed below. Also determines the following on the power curve:

- Estimated value of \( x \) in relation to \( y \) (EOX)
- Estimated value of \( y \) in relation to \( x \) (EOY)

**STATISTIC TABLE**

<table>
<thead>
<tr>
<th>Number of data items</th>
<th>CNT</th>
<th>( n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum of ( x ) data logarithmic values</td>
<td>SUMLN</td>
<td>( \Sigma lnx )</td>
</tr>
<tr>
<td>Sum of ( y ) data logarithmic values</td>
<td>SUMLNY</td>
<td>( \Sigma lny )</td>
</tr>
<tr>
<td>Sum of squares of ( x ) data logarithmic values</td>
<td>SUMLNX</td>
<td>( \Sigma lnx^2 )</td>
</tr>
<tr>
<td>Sum of squares of ( y ) data logarithmic values</td>
<td>SUMLNY</td>
<td>( \Sigma lny^2 )</td>
</tr>
<tr>
<td>Sum of products of ( x ) data logarithmic values and ( y ) data logarithmic values</td>
<td>SUMLNXLY</td>
<td>( \Sigma lnx \cdot lny )</td>
</tr>
<tr>
<td>Mean of ( x ) data logarithmic values</td>
<td>MEANLNX</td>
<td>( \Sigma lnx / n )</td>
</tr>
<tr>
<td>Mean of ( y ) data logarithmic values</td>
<td>MEANLNY</td>
<td>( \Sigma lny / m )</td>
</tr>
<tr>
<td>Population standard deviation of ( x ) data logarithmic values</td>
<td>SDLNXN</td>
<td>( \sqrt{\frac{\Sigma lnx^2 - (\Sigma lnx)^2}{n}} )</td>
</tr>
<tr>
<td>Population standard deviation of ( y ) data logarithmic values</td>
<td>SDLNYY</td>
<td>( \sqrt{\frac{\Sigma lny^2 - (\Sigma lny)^2}{m}} )</td>
</tr>
<tr>
<td>Sample standard deviation of ( x ) data logarithmic values</td>
<td>SDLNX</td>
<td>( \sqrt{\frac{\Sigma (lnx)^2 - (\Sigma lnx)^2}{n(n-1)}} )</td>
</tr>
<tr>
<td>Sample standard deviation of ( y ) data logarithmic values</td>
<td>SDLNY</td>
<td>( \sqrt{\frac{\Sigma (lny)^2 - (\Sigma lny)^2}{m(m-1)}} )</td>
</tr>
<tr>
<td>Regression constant term</td>
<td>RA</td>
<td>( a )</td>
</tr>
<tr>
<td>Regression coefficient</td>
<td>RB</td>
<td>( b )</td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>COR</td>
<td>( c )</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\text{RA} & = a = \frac{\Sigma lny - \Sigma lnx \cdot \Sigma lny}{n} \\
\text{RB} & = b = \frac{\Sigma lnx \cdot lny - \Sigma lnx \cdot \Sigma lny}{n} \\
\text{COR} & = c = \frac{n \Sigma lnx \cdot lny - (\Sigma lnx)^2 \cdot (\Sigma lny)^2}{\sqrt{(n \Sigma lnx^2 - (\Sigma lnx)^2)(n \Sigma lny^2 - (\Sigma lny)^2)}}
\end{align*}
\]

**OPERATION**

**6540**

The menu illustrated above is displayed for power regression calculations. The following seven items can be selected from this menu:

1. **I**: Data input
2. **D**: Data deletion (deletes erroneous or unnecessary data)
3. **C**: Data clear
L: Statistic display
Displays number of data items, sum of x data logarithmic values, sum of y data logarithmic values, sum of squares of x data logarithmic values, sum of squares of y data logarithmic values, sum of products of x data logarithmic values and y data logarithmic values, mean of x data logarithmic values, mean of y data logarithmic values, population standard deviation of x data logarithmic values, population standard deviation of y data logarithmic values, sample standard of x data logarithmic values, sample standard of y data logarithmic values, regression constant term, regression coefficient, and correlation coefficient.

scrolls to the following data item, to the previous data item, and or terminate statistic display.

X: Calculates x value for y on power curve.
Y: Calculates y value for x on power curve.
P: Outputs all statistics to printer.

SAMPLE

The following data for the characteristics of voltage and current for a semiconductor, form power regression, and display the statistics. Also produce an estimated value for ent at 40V.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage (x)</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>25</td>
<td>30</td>
</tr>
<tr>
<td>Current (y)</td>
<td>13</td>
<td>22</td>
<td>31</td>
<td>38</td>
<td>43</td>
</tr>
</tbody>
</table>

Regression analysis \( y = ax + b \) [y = ax \cdot b] (Data clear)
Regression analysis \( y = ax + b \) (Data clear confirmation)
Input data (x, y) \( x? \) : y? (Data input)
Input data (x, y) \( x? \) : y? (x input)
Input data (x, y) \( x? \) : y? (y input)

Remaining x, y data input) (Remaining x, y data input)
(Statistic display showing number of data and sum of x data logarithmic values) (Sum of y data logarithmic values)
(Statistical display showing number of data and sum of x data logarithmic values) (Sum of squares of x data logarithmic values)
(Statistic display showing number of data and sum of x data logarithmic values) (Sum of squares of y data logarithmic values)
(Statistic display showing number of data and sum of x data logarithmic values) (Sum of squares of y data logarithmic values)

SUMx : \( \sum x = 14.62644077 \)
SUMy : \( \sum y = 16.48676529 \)
SUMx^2 : \( \sum x^2 = 43.59315106 \)
SUMy^2 : \( \sum y^2 = 55.5545516 \)

309
SUMIN : inx = 55.30443616
SUMInxY: iny = 43.085554072

(Sum of products of x data logarithmic values and y data logarithmic values)

SUMINxY : inx = 49.65552072
MEANINX : inx/n = 2.955288155
MEANxY : inx/n = 3.2677653068
MEANINY : inx/n = 3.297753058
SDInxN : inx = 0.3679553282
SDInYN : iny/n = 0.4394431503

(Mean of x data logarithmic values)
(Mean of y data logarithmic values)
(Population standard deviation of x data logarithmic values)
(Population standard deviation of y data logarithmic values)

SDInxN : inx = 0.3679553282
SDInYN : iny/n = 0.4394431503

SDInx : inx = 0.4309431503
SDInYN : iny/n = 0.433761775

(Sample standard deviation of x data logarithmic values)
(Sample standard deviation of y data logarithmic values)

SDInx : inx = 0.433761775
SDInY : iny/n = 0.4818090693

(Regression constant term)

RA : a = 1.069436811
RB : b = 1.104376978
RD : c = 0.9942455505

(Regression coefficient)
(Correlation coefficient)

Regression analysis  (y=ax^b)

(Return to menu display)

Y

Estimation of y [y=ax^b]

Estimation of y [y=ax^b]

Estimation of y [y=ax^b]

Estimation of y [y=ax^b]

(= 62.86865253)

(Return to menu display)

Here, these data produce the power curve $y = 1.069436811 \times x^{1.104376978}$. Also, input of 40 volts results in an estimated current of 62.9mA.
EGRESSION ANALYSIS FLOWCHART (6510, 6520, 6530, 6540)
Perform estimation of the confidence interval of \( \mu \) in normal distribution \( N(\mu, \sigma^2) \) where \( \mu \) : unknown, \( \sigma^2 \) : known.

CALCULATIONS
When an \( n \)-size sample \( (x_1, x_2 \ldots x_n) \) is taken from normal distribution \( N(\mu, \sigma^2) \), the following confidence interval \( (1-\alpha) \) of confidence level for \( \mu \) is obtained:

\[
\bar{x} \pm \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{x} + \frac{\sigma}{\sqrt{n}}
\]

OPERATION

The display appears as indicated above once the library is activated. At this time, either \( Y \) or \( N \) should be pressed to perform the following procedures:

\( Y \) : New data input followed by interval estimation, additional data input, data edit, statistical check.

\( N \) : Interval estimation using previously stored data, interval estimation when data are known.

(1) \( Y \)

The menu display illustrated above appears when \( Y \) is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
(List) Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).

\( \vec{X} \) (or \( \bar{X} \)) scrolls to the following data item, \( \vec{X} \) to the previous item, and \( \vec{X} \) or \( \bar{X} \) exits the statistic display and returns to the menu.

(End) Advances to the interval estimation display (same as when N is pressed in the first step above).

\[ N \]

\[ N \backslash \vec{X} \quad \sigma^2 < \mu < \sigma^2 : \text{known} \]

\[ n = \vec{0} ? \]

(interval estimation display)

Display appears as illustrated above when the \( N \) key is pressed. The value indicated \( n \) shows the number of data currently stored in memory.

\( = 0 \): Interval estimation cannot be performed, so this should be corrected to the required data.

Number of data input (following \( Y \) above) and value of \( n \) differ: Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library gain and add, delete, or reinput data as required.

Number of data input (following \( Y \) above) matches value of \( n \): Press \( \text{EX} \)

AMPLE

The table below shows the number of customers at a store over a 5-day period. Using this data, perform interval estimation for the number of customers with a confidence level of 99%. Population standard deviation of the customers is previously known to be 120.3.

<table>
<thead>
<tr>
<th>NUMBER OF CUSTOMERS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>580</td>
<td>430</td>
<td>612</td>
<td>498</td>
<td>591</td>
</tr>
</tbody>
</table>

\[ N \backslash \vec{X} \quad \sigma^2 < \mu < \sigma^2 : \text{known} \]

\[ \text{input new data (Y/N)?} \]

\[ \text{Input data (x)} \]

\[ > \text{Input Delete Clear List End ?} \]

\[ \text{Input data (x)} \]

\[ > \text{Input Delete Clear List End ?} \]

\[ \text{Input data (x)} \]

\[ (\text{EXE}) \text{: menu} \]

\[ x ? \]

\[ \text{Input data (x)} \]

\[ (\text{EXE}) \text{: menu} \]

\[ x ? \]

\[ \text{Input data (x)} \]

\[ (\text{EXE}) \text{: menu} \]

\[ x ? \]

\[ \text{Input data (x)} \]

\[ > \text{Input Delete Clear List End ?} \]

\[ N \backslash \vec{X} \quad \sigma^2 < \mu < \sigma^2 : \text{known} \]

\[ n = 5 ? \]

\[ \text{Input data (x)} \]

\[ > \text{Input Delete Clear List End ?} \]

\[ N \backslash \vec{X} \quad \sigma^2 < \mu < \sigma^2 : \text{known} \]

\[ x = 612 \backslash 498 \]

\[ \text{Input data (x)} \]

\[ > \text{Input Delete Clear List End ?} \]

\[ N \backslash \vec{X} \quad \sigma^2 < \mu < \sigma^2 : \text{known} \]

\[ x = 591 \]

\[ \text{(Select new data input.)} \]

\[ \text{(Select data clear.)} \]

\[ \text{(Data cleared.)} \]

\[ \text{(Select data input.)} \]

\[ \text{(Enter first data item.)} \]

\[ \text{(Enter remaining data items.)} \]

\[ \text{(Return to menu.)} \]

\[ \text{(Select End to proceed to interval estimation.)} \]

\[ \text{(Press EX after checking number of data items.)} \]

\[ \text{(Press EX after checking mean.)} \]
Here, it is determined that the mean for number of customers $\mu$ with a confidence level of 99% is $403.6 < \mu < 680.8$.

MEAN INTERVAL ESTIMATION FLOWCHART (FOR KNOWN VARIANCE)
MEAN INTERVAL ESTIMATION
(FOR UNKNOWN VARIANCE)

performs estimation of the confidence interval of \( \mu \) in normal distribution \( N(\mu, \sigma^2) \); where \( \mu \) is unknown, \( \sigma^2 \) is unknown.

CALCULATIONS

when an \( n \)-size sample \((x_1, x_2 \cdots x_n)\) is taken from normal distribution \( N(\mu, \sigma^2) \),

\[
\bar{x} - t\left(\frac{\sigma}{\sqrt{n}}, n-1\right) \sqrt{\frac{V}{n}} < \mu < \bar{x} + t\left(\frac{\sigma}{\sqrt{n}}, n-1\right) \sqrt{\frac{V}{n}}
\]

obtained in accordance with degree of freedom \((n-1)\) of the \( t \)-distribution.

\[
V = \frac{\sum(x - \bar{x})^2}{n-1}
\]

\( \bar{x} \): population mean
\( \sigma^2 \): population variance
\( \alpha \): significance level
\( \bar{x} \): sample mean
\( V \): unbiased variance
\( 1 - \alpha \): confidence level

ERATION

120
320

The display appears as indicated above once the library is activated. At this time, either \( Y \) or \( N \) should be pressed to perform the following procedures:

New data input followed by interval estimation, additional data input, data edit, statistic check.
Interval estimation using previously stored data, interval estimation by inputting each value.

\( Y \)

The menu display illustrated above appears when \( Y \) is pressed. One of the following characters is then pressed to perform the corresponding function.

\((\text{Input})\) : Data input (for input or addition of data).
\((\text{Delete})\) : Data delete (for deletion of erroneous or unnecessary data).
\((\text{Clear})\) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
U (or EC) scrolls to the following data item, C to the previous item, and EC or EC exits the statistic display and returns to the menu.
E (End) : Advances to the interval estimation display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. The value indicated for n shows the number of data currently stored in memory.

* n = 0: Interval estimation cannot be performed, so this should be corrected to the required data.
* Number of data input (following Y above) and value of n differ: Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.

**Number of data input (following Y above) matches value of n:** Press EC.

**EXAMPLE**

The table below shows the number of customers over a 5-day period for five drugstores selected at random in a certain area. Using this data, perform interval estimation for the number of customers at one drugstore with a confidence level of 95%.

<table>
<thead>
<tr>
<th>NUMBER OF CUSTOMERS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>245</td>
<td>366</td>
<td>271</td>
<td>493</td>
<td>306</td>
<td></td>
</tr>
</tbody>
</table>

N (s, c) b<s<b
input new data (Y/N)? Y
input data (x)
> input, delete, clear, list, end? C
input data (x)
> input, delete, clear, list, end? Y
input data (x)
> input, delete, clear, list, end? T
input data (x) [EXE]: menu
x? _
input data (x) [EXE]: menu
245 EC
input data (x) [EXE]: menu
366 ex 271 EC 493 EC 306 EC...
input data (x) [EXE]: menu
input data (x)
> input, delete, clear, list, end? EC
input data (x)
> input, delete, clear, list, end? E
N (s, c) b<s<b
n = 5 _
N (s, c) b<s<b
x = 336.2 _
N (s, c) b<s<b
v = 9738.7 _

(Select new data input.)
(Select data clear.)
(Select data input.)
(Enter first data item.)
(Enter remaining data items.)
(Return to menu)
(Select End to proceed to interval estimation.)
(Press EC after checking number of data items.)
(Press EC after checking mean.)
Here, it is determined that the mean for number of customers $\mu$ with a confidence level of $1 - \alpha$ is $213.7 < \mu < 458.7$.

**AN INTERVAL ESTIMATION FLOWCHART (FOR UNKNOWN VARIANCE)**
VARIANCE INTERVAL ESTIMATION

Performs estimation of the confidence interval of $\sigma^2$ in normal distribution $N(\mu, \sigma^2)$; where $\mu$ : unknown, $\sigma^2$ : unknown).

CALCULATIONS
When an n-size sample $(x_1, x_2, \ldots, x_n)$ is taken from normal distribution $N(\mu, \sigma^2)$, the confidence interval of the confidence level $(1-\alpha)$ of $\sigma^2$ is obtained by

$$\frac{S}{\chi^2\left(\frac{\alpha}{2}, n-1\right)} < \sigma^2 < \frac{S}{\chi^2\left(1-\frac{\alpha}{2}, n-1\right)}$$

in accordance with $\chi^2$ distribution of the degree of freedom $(n-1)$.

\[ S = \sum (x - \bar{x})^2 \]

OPERATION

6630 [LB]

\[ N(\mu, \sigma^2) \quad a < \sigma^2 < b \]

input new data (Y/N)?

The display appears as indicated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by interval estimation, additional data input, data edit, statistic check.
N : Interval estimation using previously stored data, interval estimation by inputting each value.

(1) Y

Input data (x)

>Input, Delete, Clear, List, End ?

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
(or ) scrolls to the following data item, to the previous item, and or exits the statistic display and returns to the menu.
E (End) : Advances to the interval estimation display (same as when N is pressed in the first step above).

(2) N

N

The display appears as illustrated above when the key is pressed. The value indicated for shows the number of data currently stored in memory.

• : Interval estimation cannot be performed, so this should be corrected to the required data.
• Number of data input (following above) and value of differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
• Number of data input (following above) matches value of : Press .

EXAMPLE

The table below shows the number of pins contained in five different boxes of the same size produced by the same manufacturer. Using this data, perform interval estimation with 99% confidence level for the pin variance.

<table>
<thead>
<tr>
<th>NUMBER OF PINS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>99</td>
<td>100</td>
<td>103</td>
<td>101</td>
<td></td>
</tr>
</tbody>
</table>

N

The table above shows the number of pins contained in five different boxes of the same size produced by the same manufacturer. Using this data, perform interval estimation with 99% confidence level for the pin variance.

N

The display appears as illustrated above when the key is pressed. The value indicated for shows the number of data currently stored in memory.

• : Interval estimation cannot be performed, so this should be corrected to the required data.
• Number of data input (following above) and value of differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
• Number of data input (following above) matches value of : Press .

EXAMPLE

The table below shows the number of pins contained in five different boxes of the same size produced by the same manufacturer. Using this data, perform interval estimation with 99% confidence level for the pin variance.

<table>
<thead>
<tr>
<th>NUMBER OF PINS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>99</td>
<td>100</td>
<td>103</td>
<td>101</td>
<td></td>
</tr>
</tbody>
</table>

(Select new data input.)
(Select data clear.)
(Data cleared.)
(Select data input.)
(Enter first data item.)
(Enter remaining data items.)
(Return to menu.)
Here, it is determined that the variance of the number of pins $\sigma^2$ with a confidence level of 99% is $0.6729 < \sigma^2 < 48.31$.

VARIANCE INTERVAL ESTIMATION FLOWCHART
performs estimation of the confidence interval of \( \sigma \) in normal distribution \( N (\mu, \sigma^2) \); where \( \mu \) : unknown, \( \sigma^2 \) : unknown.

**Calculations**

When an \( n \)-size sample \((x_1, x_2, \ldots, x_n)\) is taken from normal distribution \( N (\mu, \sigma^2) \), the confidence interval of the confidence level \((1 - \alpha)\) of \( \sigma^2 \) is obtained by

\[
\frac{S}{\chi^2(\frac{\alpha}{2}, n-1)} < \sigma < \sqrt{\frac{S}{\chi^2(1 - \frac{\alpha}{2}, n-1)}}
\]

acccordance with the \( \chi^2 \) distribution of the degree of freedom \((n - 1)\).

\[
\begin{align*}
&\frac{S}{\chi^2(\frac{\alpha}{2}, n-1)} < \sigma < \sqrt{\frac{S}{\chi^2(1 - \frac{\alpha}{2}, n-1)}} \\
&\text{\( \mu \): population mean} \\
&\text{\( \sigma^2 \): population variance} \\
&\text{\( \alpha \): significance level} \\
&\text{\( S \): sum of squares} \\
&\text{\( 1 - \alpha \): confidence level} \\
&\text{\( S = \sum (x - \bar{x})^2 \)}
\end{align*}
\]

**Eratation**

**40**

**LIB**

\[
N(\mu, \sigma^2) \quad \text{[NO]} \quad \text{INPUT NEW DATA (Y/N)?}
\]

The display appears as indicated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

- New data input followed by interval estimation, additional data input, data edit, statistic check.
- Interval estimation of previously stored data, interval estimation by inputting each value.

**Y**

**INPUT DATA (X)**

**INPUT DELETE/CLEAR/LIST/END?**

The menu display illustrated above appears when \( Y \) is pressed. One of the following character is then pressed to perform the corresponding function.

- **Input** : Data input (for input or addition of data).
- **Delete** : Data delete (for deletion of erroneous or unnecessary data).
- **Clear** : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
   X or Y scrolls to the following data item, X to the previous item, and X or Y exits the statistic display and returns to the menu.
E (End) : Advances to the interval estimation display (same as when N is pressed in the first step above).

(2) N

| N(p,e) a < e < b
| n = 5 ?

(Interval estimation display)

The display appears as illustrated above when the N key is pressed. The value indicated for n shows the number of data currently stored in memory.

- n = 0 : Interval estimation cannot be performed, so this should be corrected to the required data.
- Number of data input (following Y above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
- Number of data input (following Y above) matches value of n : Press EXE.

EXAMPLE
The table below shows the measured volume of the contents of five different randomly selected cans of a soft drink produced by the same manufacturer. Using this data, perform interval estimation with 99% confidence level for the sample standard deviation of the content volume.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOLUME</td>
<td>1.20</td>
<td>1.08</td>
<td>1.15</td>
<td>1.22</td>
<td>1.17</td>
</tr>
</tbody>
</table>

N(p,e) a < e < b
input new data (Y/N) ?
Y
input data (x)
> input. delete. clear. list. end ? _
C
input data (x)
clear data (y/n) ?
Y
input data (x)
> input. delete. clear. list. end ? _
L
input data (x) :
EXE ; menu
x?

1.20 EXE

1.08 EXE 1.15 EXE 1.22 EXE 1.17 EXE

input data (x) :
EXE ; menu
x?

(Enter first data item.)

(Retrieve previous menu.)

(Retrieve previous menu.)

(Select End to proceed to interval estimation)

(Select new data input.)

(Enter remaining data items.)

(Select data clear.)

(Data cleared.)

(Select data input.)

(Enter remaining data items.)
Here, it is determined that the sample standard deviation of the volume of the cans' contents $\sigma$ with a confidence level of 99% is $0.02808 < \sigma < 0.238$.

**STANDARD DEVIATION INTERVAL ESTIMATION FLOWCHART**
VARIANCE RATIO INTERVAL ESTIMATION

Performs estimation of the confidence interval of \( \frac{\sigma_1^2}{\sigma_2^2} \) for the two normal distributions \( N(\mu_1, \sigma_1^2) \) and \( N(\mu_2, \sigma_2^2) \), where \( \mu_1, \sigma_1^2, \mu_2 \) and \( \sigma_2^2 \) are all unknown.

**CALCULATIONS**

When \( n_1 \)-size sample \( x_1 (x_{11}, x_{12}, \ldots, x_{1n_1}) \) is taken from normal distribution \( N(\mu_1, \sigma_1^2) \), and \( n_2 \)-size sample \( x_2 (x_{21}, x_{22}, \ldots, x_{2n_2}) \) is taken from normal distribution \( N(\mu_2, \sigma_2^2) \), the confidence interval of the confidence level \((1 - \alpha)\) of \( \frac{\sigma_1^2}{\sigma_2^2} \) is obtained by

\[
\frac{V_1}{V_2} \cdot \frac{1}{F\left(\frac{\sigma_1^2}{\sigma_2^2}, \frac{n_1}{n_2} - 1, n_1 - 1\right)} < \frac{\sigma_1^2}{\sigma_2^2} < \frac{V_2}{V_1} \cdot F\left(\frac{\sigma_2^2}{\sigma_1^2}, \frac{n_2}{n_1} - 1, n_2 - 1\right)
\]

in accordance with the \( F \) distribution of the degrees of freedom \((n_1 - 1, n_2 - 1)\).

\[
F = \sum \frac{(x - \bar{x})^2}{S(x - \bar{x})^2}
\]

\( x_1, x_2 \) : population means
\( \sigma_1, \sigma_2 \) : population variances
\( \alpha \) : significance level
\( V_1, V_2 \) : unbiased variances
\( 1 - \alpha \) : confidence level

**OPERATION**

**6650**

\[ N(i, 1, c_1^2), N(i, 2, c_2^2), B < c_2^2/c_1^2 < D \]

**INPUT NEW DATA X1 (Y/N)?**

The display appears as indicated above once the library is activated. At this time, either **Y** or **N** should be pressed to perform the following procedures:

**Y** : New data input followed by interval estimation, additional data input, data edit, statistic check.

**N** : Interval estimation using previously stored data, interval estimation by inputting each value.

**(1) Y**

\[ \text{Input data (x1)} \]

\[ \text{Input, Delete, Clear, List, End?} \]

The menu display illustrated above appears when **Y** is pressed. One of the following character keys is then pressed to perform the corresponding function.
input) : Data input (for input or addition of data).
Delete) : Data delete (for deletion of erroneous or unnecessary data).
Clear) : Data clear (for deletion of previously stored data. This operation also clears
statistics).
List) : Statistic display (for display of number of data items, sum, sum of squares,
mean, population standard deviation, sample standard deviation).
N (or R) scrolls to the following data item, to the previous item, and C
or D exits the statistic display and returns to the menu.
End) : Advances to the interval estimation display (same as when N is pressed in
the first step above).

Note that the data input referred to here is for data items $x_1$ through $x_{1n}$.

\[
N
\]

\[
\frac{N(x_1, c)^2}{N(x_2, c)^2} \leq c^2/c_1^2 < 1
\]

The display appears as illustrated above when the N key is pressed. Note that this display
most identical to the initial display which appears immediately after entering library opera-
tions. The difference, however, is that the question concerning new data input here is for
items $x_1$ through $x_2$, while the data input being queried on the original display is for
items $x_1$ through $x_{1n}$.

Y)

The result as that produced by pressing Y in step (1) above. Note, however, that the data
entered or corrected here is $x_{21}$ through $x_{2n}$.

N)

\[
\frac{N(x_1, c)^2}{N(x_2, c)^2} \leq c^2/c_1^2 < 1
\]

The display appears as illustrated above when the N key is pressed. The value indicated
shows the number of $x_1$ ($x_1 - x_{1n}$) data currently stored in memory.

= 0 : Interval estimation cannot be performed, so this should be corrected to the
required data.

Number of data input (following Y above) and value of n differ : Confirm that some data
have not been omitted during the input or that two or more data items have been input
gether for a single entry. In either case, terminate the library operation. Enter the library
again and add, delete, or reinput data as required.

Number of data input (following Y above) matches value of n : Press C.

Then C is pressed, a display similar to that above is produced for $x_2$ ($x_2 - x_{2n}$) data
items. After confirmation and/or corrections as described in (2-2), press C to continue.

\textbf{Example}

The table below shows the measured diameters of ten randomly selected ball bearings. The
company producing the ball bearings uses two separate production lines (A and B), so five
com samples were taken from each line. Using this data, perform interval estimation with
confidence level for the variance ratio of the diameters.

\begin{center}
\begin{tabular}{|c|c|c|c|c|c|}
\hline
\textbf{Diameter} & 1 & 2 & 3 & 4 & 5 \\
\hline
A & 1.01 & 1.00 & 1.01 & 1.02 & 1.00 \\
B & 1.00 & 0.99 & 1.01 & 1.00 & 0.98 \\
\hline
\end{tabular}
\end{center}
Here, it is determined that the variance ratio between the two lines with a confidence level of 95% is $0.1934 < \frac{\sigma_2^2}{\sigma_1^2} < 17.84$. 
Performs estimation of the confidence interval $\mu_1 - \mu_2$ for two equal distributions $N(\mu_1, \sigma^2)$ and $N(\mu_2, \sigma^2)$, where $\mu_1$, $\mu_2$ and $\sigma^2$ are all unknown.

**CALCULATIONS**

When $n_1$-size sample $x_1$ ($x_{11}, x_{12}, ..., x_{1n_1}$) is taken from normal distribution $N(\mu_1, \sigma^2)$, and $n_2$-size sample $x_2$ ($x_{21}, x_{22}, ..., x_{2n_2}$) is taken from normal distribution $N(\mu_2, \sigma^2)$, the confidence interval of the confidence level $(1 - \alpha)$ of $\mu_1 - \mu_2$ is obtained by

$$
\bar{x}_1 - \bar{x}_2 - t\left(\frac{\alpha}{2}, n_1 + n_2 - 2\right) \sqrt{\frac{1}{n_1} + \frac{1}{n_2} \left(\frac{S_1 + S_2}{n_1 + n_2 - 2}\right)} < \mu_1 - \mu_2 < \bar{x}_1 - \bar{x}_2 + t\left(\frac{\alpha}{2}, n_1 + n_2 - 2\right)
$$

where

$$
\bar{x}_1, \bar{x}_2 : \text{sample means}
$$

in accordance with t-distribution of the degree of freedom $(n_1 + n_2 - 2)$.

**OPERATION**

6660 UB

The display appears as indicated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

**Y:** New data input followed by interval estimation, additional data input, data edit, statistic check.

**N:** Interval estimation of previously stored data, interval estimation by inputting each value.

(1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
X (or X) scrolls to the following data item, C to the previous item, and E or E exits the statistic display and returns to the menu.
E (End) : Advances to the interval estimation display (same as when N is pressed in the first step above).

* Note that the data input referred to here is for data items x_{1i} through x_{1ni}.

(2) N

\[
\begin{array}{c}
N(x: \sigma^2), N(x: \mu, \sigma^2) \quad \theta < \mu - \mu \leq \theta
\end{array}
\]

Input new data xe (Y/N) ?

The display appears as illustrated above when the N key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items x_{12} through x_{22n}, while the data input being queried on the original display is for data items x_{1i} through x_{1ni}.

(2-1) Y

Same result as that produced by pressing Y in step (1) above. Note, however, that the data being entered or corrected here is x_{2i} through x_{22n}.

(2-2) N

\[
\begin{array}{c}
N(x: \sigma^2), N(x: \mu, \sigma^2) \quad \theta < \mu - \mu \leq \theta
\end{array}
\]

n_{1i} = 5 ?

(Number of data display)

The display appears as illustrated above when the N key is pressed. The value indicated for n shows the number of x_{1} (x_{11} - x_{1ni}) data currently stored in memory.

- n_{1} = 0 : Interval estimation cannot be performed, so this should be corrected to the required data.
- Number of data input (following Y above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
- Number of data input (following Y above) matches value of n : Press X.

When X is pressed, a display similar to that above is produced for x_{2} (x_{21} - x_{22n}) data items. After confirmation and/or corrections as described in (2-2), press X to continue.

**EXAMPLE**

The table below shows a comparison of the production volume for a factory for two consecutive weeks. Using this data, perform interval estimation with 95% confidence level for the difference in the mean for the two weeks.

<table>
<thead>
<tr>
<th>PRODUCTION VOLUME (1)</th>
<th>MON</th>
<th>TUE</th>
<th>WED</th>
<th>THU</th>
<th>FRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>WEEK 1</td>
<td>53</td>
<td>59</td>
<td>56</td>
<td>60</td>
<td>54</td>
</tr>
<tr>
<td>WEEK 2</td>
<td>55</td>
<td>62</td>
<td>60</td>
<td>61</td>
<td>58</td>
</tr>
</tbody>
</table>

329
Here, it is determined that the difference in means $\mu_1 - \mu_2$ between the two weeks with a confidence level of 95% is $-7.052 < \mu_1 - \mu_2 < 1.452$
RATIO INTERVAL ESTIMATION

Performs estimation of the confidence interval \( p \) for binomial distribution \( B(1, p) \).

**CALCULATIONS**

When \( n \)-size sample \((x_1, x_2 \ldots x_n)\) is taken from binomial distribution \( B(1, p) \), the confidence interval of the confidence level \((1 - \alpha)\) of \( p \) is obtained by

\[
\frac{\hat{x}}{n} - \left( \frac{\alpha}{2} \right) \sqrt{\frac{1}{n} \left( \frac{\hat{x}}{n} (1 - \frac{\hat{x}}{n}) \right)} < p < \frac{\hat{x}}{n} + \left( \frac{\alpha}{2} \right) \sqrt{\frac{1}{n} \left( \frac{\hat{x}}{n} (1 - \frac{\hat{x}}{n}) \right)}
\]

in accordance with an approximation of the standard normal distribution \( N(0, 1^2) \).

**OPERATION**

6670 UB

\[
\text{B(1, p) } a < p < b \quad n = 0 \ ? -
\]

**EXAMPLE**

12 defects are found for 1000 bolts produced by a certain factory. Using this data, perform interval estimation with 99% confidence level for the defect rate of the bolts.

\[
\begin{align*}
\text{B(1, p) } a & < p < b \\
n & = 1000 \ ? -
\end{align*}
\]

(Enter total number of samples.)

\[
\begin{align*}
\text{B(1, p) } a & < p < b \\
12 & = 0 \ ? -
\end{align*}
\]

(Enter number of defects.)

\[
\begin{align*}
\text{Confidence level } (1 - \alpha)(\%) \\
99 & = 0.05 \ ? -
\end{align*}
\]

(Enter confidence level.)

\[
\begin{align*}
\text{B(1, p) } a & < p < b \\
0.003191 & < p < 0.02087 \quad n = 1000 \ ? -
\end{align*}
\]

Here, it is determined that the defect rate \( p \) for the bolts with a confidence level of 99% is \( 0.003191 < p < 0.02087 \).
RATIO DIFFERENCE INTERVAL ESTIMATION

forms estimation of the confidence interval $p_1 - p_2$ for two binomial distributions $B(1, p_1)$ and $B(1, p_2)$.

CALCULATIONS

When $n_1$-size sample $x_1 (x_{11}, x_{12}, \ldots, x_{1n_1})$ is taken from binomial distribution $B(1, p_1)$, and $n_2$-size sample $x_2 (x_{21}, x_{22}, \ldots, x_{2n_2})$ is taken from binomial distribution $B(1, p_2)$, the confidence interval of the confidence level $(1 - \alpha)$ of $p_1 - p_2$ is obtained by

$$\left( \frac{\sum x_1}{n_1} - \frac{\sum x_2}{n_2} \right) \pm \left( \frac{\sigma}{\sqrt{n}} \right) \sqrt{\frac{1}{n_1} \left( \frac{\sum x_1}{n_1} (1 - \frac{\sum x_1}{n_1}) \right) + \frac{1}{n_2} \left( \frac{\sum x_2}{n_2} (1 - \frac{\sum x_2}{n_2}) \right)} < p_1 - p_2 < \left( \frac{\sum x_1}{n_1} - \frac{\sum x_2}{n_2} \right)$$

according to an approximation of the standard normal distribution $N(0, 1^2)$.

![Normal distribution curve N(0, 1^2)](image)

- $\mu$: mean
- $\sigma$: standard deviation
- $\alpha$: significance level
- $1 - \alpha$: confidence level

ERATION

880 [UB] $E(1, p_1) \cdot B(1, p_2) \quad \alpha < p_1 - p_2 < \beta$

SAMPLE

The table below shows a comparison of the number of defects for a factory for two consecutive months. Using this data, perform interval estimation with 95% confidence level for the difference in the rates of defect.

<table>
<thead>
<tr>
<th></th>
<th>Finished products</th>
<th>Number of defects</th>
</tr>
</thead>
<tbody>
<tr>
<td>MONTH 1</td>
<td>1500</td>
<td>23</td>
</tr>
<tr>
<td>MONTH 2</td>
<td>1200</td>
<td>15</td>
</tr>
<tr>
<td>1500</td>
<td>( n_1 = 6 )</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>( n_2 = 0 )</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>( n_3 = 0 )</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>( n_4 = 0 )</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>( n_5 = 0 )</td>
<td></td>
</tr>
</tbody>
</table>

Here, it is determined that the difference in probabilities \( p_1 - p_2 \) between the two months with a confidence level of 95% is \(-0.006009 < p_1 - p_2 < 0.01168\).
forms hypothesis testing of \( \mu \) in normal distribution \( N(\mu, \sigma^2) \) where \( \mu \) : unknown, \( \sigma^2 \) : unknown.

**CALCULATIONS**

An n-size sample \( (x_1, x_2 \ldots x_n) \) is taken from normal distribution \( N(\mu, \sigma^2) \). At this time, critical regions are established on both sides of the normal distribution as shown in the illustration when:

- null hypothesis (Null hypothesis) \( H_0 : \mu = \mu_0 \)
- alternative hypothesis \( H_1 : \mu \neq \mu_0 \)

A test is performed using

\[
\frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}} > \nu\left(\frac{\alpha}{2}\right)
\]

\( \bar{x} \) : sample mean

\( \mu_0 \) : population mean

\( \sigma^2 \) : population variance

\( e \) : population standard deviation

\( x \) : sample mean

\( \alpha \) : significance level

**ERATION**

10 **LIB**

**Test**

**Input new data (Y/N) ?**

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

- New data input followed by test, additional data input, data edit, statistic check.
- Test of previously stored data, test by inputting each value.

**Y**

**Input data (x)**

**Input Day. Clear. List. End ?**

The menu display illustrated above appears when Y is pressed. One of the following characters is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics operations).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
X (or  ) scrolls to the following data item,  to the previous item, and  or  exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

\[
\begin{array}{ccc}
\text{Test} & \alpha = 0 & \beta = 0 \\
\end{array}
\]

(Test display)

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

The table below shows the measured speed of five new football players over 100 meters. These times will be used to determine whether or not these players meet the team standards. Perform a test on the data with a significance level of 5%. The mean time for the entire team is 11.4 seconds, with a standard deviation of 1.30.

<table>
<thead>
<tr>
<th>TIME</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12.3</td>
<td>11.6</td>
<td>10.9</td>
<td>12.8</td>
<td>11.4</td>
</tr>
</tbody>
</table>

\[
\begin{array}{ccc}
\text{Y} & \text{Input data (x)} & \text{Input data (x)} \text{new data (Y/N) ?} \\
\end{array}
\]

(Select new data input.)

\[
\begin{array}{ccc}
\text{Y} & \text{Input data (x)} & \text{Input data (x)} \text{clear data (Y/N) ?} \\
\end{array}
\]

(Select data clear.)

\[
\begin{array}{ccc}
\text{Y} & \text{Input data (x)} & \text{Input data (x)} \text{new data (Y/N) ?} \\
\end{array}
\]

(Data cleared.)

\[
\begin{array}{ccc}
\text{Y} & \text{Input data (x)} & \text{Input data (x)} \text{new data (Y/N) ?} \\
\end{array}
\]

(Select data input.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Enter first data item.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Enter remaining data items.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Return to menu.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Select End to proceed to test.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Enter mean.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Enter population standard deviation.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Press  after checking number of data.)

\[
\begin{array}{ccc}
\text{Y} & \text{Test n = 5} & \alpha = 0 \\
\end{array}
\]

(Enter significance level s(%) after checking data mean.)
a, it is determined that the speeds of the new players meet the team standards. In this example, the number of data items was limited to five for ease of understanding. In actual practice, a small number of data may cause erroneous results (standard: n \geq 50).

Population Mean Test Flowchart (Two-Sided for Known Variance)
POPULATION MEAN TEST
(RIGHT SIDED) : FOR KNOWN VARIANCE

Performs hypothesis testing of $\mu$ in normal distribution $N(\mu, \sigma^2)$, where $\mu$: unknown, $\sigma^2$: known.

CALCULATIONS

An n-size sample $(x_1, x_2 \ldots x_n)$ is taken from normal distribution $N(\mu, \sigma^2)$. At this time, the critical region is established on the right side of the normal distribution as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) $H_0: \mu = \mu_0$
Alternative hypothesis $H_1: \mu > \mu_0$

The test is performed using

$$\frac{\bar{x} - \mu_0}{\sigma} > \Phi (\alpha)$$

Normal distribution curve

$\bar{x}$: sample mean
$\mu_0$: population mean
$\sigma$: population standard deviation
$\phi(\alpha)$: significance level

OPERATION

6711

Test $H_0: \mu = \mu_0$, $H_1: \mu > \mu_0$
Input new data (Y/N) ?

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y: New data input followed by test, additional data input, data edit, statistic check.
N: Test of previously stored data, test by inputting each value.

(1) Y

Input data $(x)$ Input Delete Clear List End ?

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.
(Input) : Data input (for input or addition of data).
(Delete) : Data delete (for deletion of erroneous or unnecessary data).
(Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
(List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
( ) or ( ) scrolls to the following data item, ( ) to the previous item, and ( ) or ( ) exits the statistic display and returns to the menu.
(End) : Advances to the test display (same as when N is pressed in the first step above).

Test
H₀ : p = p₀  H₁ : p > p₀

A display appears as illustrated above when the (N) key is pressed. From this point, various statistical values are entered for the test.

AMPLE

A factory is considering replacing 50 obsolete machines with newer models. Management suspects, however, that the capacity of the new machines are the same as those currently in use. The data included in the table below are the results of tests performed on five units of the new machines. Using these results, determine whether or not the capacity of the new machines is equal to the existing machines by performing a test on the data with a significance level of 5%. The capacity of the exiting machines is 432 units/hour, with a standard deviation of 15.

<table>
<thead>
<tr>
<th>UNITS/HOUR</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>475</td>
<td>501</td>
<td>483</td>
<td>492</td>
<td>487</td>
</tr>
</tbody>
</table>

Test
H₀ : p = p₀  H₁ : p > p₀

Input new data (Y/N) ?

Input data (x)
>Input, Delete, Clear, List, End ?

Input data (x)
Clear data (Y/N) ?

Input data (x)
>Input, Delete, Clear, List, End ?

Input data (x) [EXE]: menu
x?

Input data (x) [EXE]: menu
x?

Input data (x) [EXE]: menu
x?

Test
H₀ : p = p₀  H₁ : p > p₀ (Select End)

Test
c = 0 ?

Test
n = 5 ?
Here, it is determined that it cannot be said that the capacity of the new machines are identical to that of the existing machines. The new machines have higher capacities. In this example, the number of data items was limited to five for ease of understanding. In actual tests, smaller number of data may cause erroneous results (standard: n ≥ 50).

POPULATION MEAN TEST FLOWCHART (RIGHT SIDED FOR KNOWN VARIANCE)
forms hypothesis testing of $\mu$ in normal distribution $N(\mu, \sigma^2)$ where $\mu$ : unknown, $\sigma^2$ : unknown.

CALCULATIONS

$n$-size sample ($x_1, x_2 \ldots x_n$) is taken from normal distribution $N(\mu, \sigma^2)$. At this time, the critical region is established on the left side of the normal distribution as shown in the illustration when:

- Hypothesis to be tested (Null hypothesis) $H_0 : \mu = \mu_0$
- Alternative hypothesis $H_1 : \mu < \mu_0$

The test is performed using

$$\frac{\bar{x} - \mu_0}{\sigma \over \sqrt{n}} < -\nu(\alpha)$$

$\bar{x}$ : sample mean
$\mu_0$ : population mean
$\sigma$ : population standard deviation
$\nu(\alpha)$ : significance level

OPERATION

712 [8]

The display appears as illustrated above once the library is activated. At this time, either $Y$ or $N$ should be pressed to perform the following procedures:

- New data input followed by test, additional data input, data edit, statistic check.
- Test of previously stored data, test by inputting each value.

1) $Y$

The menu display illustrated above appears when $Y$ is pressed. One of the following characters is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List)  : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
[ or ] scrolls to the following data item, [ to the previous item, and [ or ] exits the statistic display and returns to the menu.
E (End)  : Advances to the test display (same as when N is pressed in the first step above).

(2) N

Test
H_o : \mu = 0
H_1 : \mu < 0

N

(Select new data input.)
Input data (x)
> Input, Delete, Clear, List, End ?

C

(Select data clear.)
Input data (x)
clear data (Y/N) ?

Y

(Data cleared.)
Input data (x)
> Input, Delete, Clear, List, End ?

T

(Select data input.)
Input data (x) [EXE]: menu

1229 EXE

(Enter first data item.)

1201 EXE 1234 EXE 1225 EXE 1247 EXE

(Enter remaining data items.)

Input data (x) [EXE]: menu

(Enter population mean.)

1234 EXE

(Select End.)

Test
\mu = 0 ?
H_o : \mu = 0
H_1 : \mu < 0

n = 5 ?

(Enter population standard deviation.)

7.6 EXE

(Test
\mu = 1227.2 ?
H_o : \mu = 0
H_1 : \mu < 0

Press EXE after checking number of data.)

The display appears as illustrated above when the N key is pressed. From this point, various statistical values are entered for the test.

EXAMPLE

A company is considering replacing 500 lights has been approached by a salesman who claims to have much lower price, but with a comparable service life. The data included in the table below are the results of tests performed on five units of the new lights. Using these results, determine whether or not the life expectancy of these lights is equal to the existing lights by performing a test on the data with a significance level of 5%. The mean service life of the existing lights is 1,234 hours/light, with a standard deviation of 7.6.

<table>
<thead>
<tr>
<th>SERVICE LIFE (HOURS)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1229</td>
<td>1201</td>
<td>1234</td>
<td>1225</td>
<td>1247</td>
</tr>
</tbody>
</table>
Here, it is determined that it cannot be said that the service life of the cheaper lights is identical to that of the existing lights. The cheaper lights have shorter lives. In this example, the number of data items was limited to five for ease of understanding. In actual tests, lower number of data may cause erroneous results (standard: \( n \geq 50 \)).

**POPULATION MEAN TEST FLOWCHART (LEFT SIDED FOR KNOWN VARIANCE)**

- Library start
- New data input
  - Menu 1
    - Process selection
      - Menu 2
        - Data input
          - Only?
            - Yes
            - No
          - Input of data to be deleted
            - Only?
              - Yes
              - No
          - Data clear
            - Process selection
              - Data clear
              - First statistic display
                - Statistic display
                  - Only?
                    - Yes
                    - No
                  - Next statistic display
                    - Statistic present?
                      - Yes
                      - No
POPULATION MEAN TEST
(TWO-SIDED): FOR UNKNOWN VARIANCE

Performs hypothesis testing of \( \mu \) in normal distribution \( N(\mu, \sigma^2) \) where \( \mu \) : unknown, \( \sigma^2 \) : unknown.

CALCULATIONS
An n-size sample \((x_1, x_2, \ldots, x_n)\) is taken from normal distribution \( N(\mu, \sigma^2) \). At this time, critical regions are established on both sides of t-distribution in accordance with the t-distribution of the degree of freedom \((n-1)\) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0: \mu = \mu_0 \)

Alternative hypothesis \( H_1: \mu \neq \mu_0 \)

The test is performed using

\[
\left| \frac{\bar{x} - \mu_0}{\sqrt{\frac{s^2}{n}}} \right| > t \left( \frac{\alpha}{2}, n-1 \right)
\]

\( t \)-distribution of degree of freedom \((n-1)\)

\[ s^2 = \frac{\sum (x - \bar{x})^2}{n-1} \]

(\( s^2 \): unbiased variance
\( \bar{x} \): sample mean
\( \alpha \): significance level)

OPERATION

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y: New data input followed by test, additional data input, data edit, statistic check.
N: Test of previously stored data, test by inputting each value.

1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
        (or EXEC) scrolls to the following data item, (or EXEC) to the previous item, and or EXEC exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

The following data represent test scores for a group of students. The same test has been conducted more than one hundred times in the past, with a mean score of 5.4. Use the data to determine whether or not the scores for this group of students are equivalent to past scores with a significance level of 5%.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA</td>
<td>5.3</td>
<td>6.2</td>
<td>6.0</td>
<td>5.8</td>
<td>5.5</td>
</tr>
</tbody>
</table>

The display appears as above when the N key is pressed. From this point, various parameters are entered for the test.
Here, the test results can be said to be equivalent.

**POPULATION MEAN TEST FLOWCHART (TWO-SIDED FOR UNKNOWN VARIANCE)**
Performs hypothesis testing of $\mu$ in normal distribution $N(\mu, \sigma^2)$ where $\mu$ : unknown, $\sigma^2$ : unknown).

**CALCULATIONS**

An n-size sample $(x_1, x_2 \cdots x_n)$ is taken from normal distribution $N(\mu, \sigma^2)$. At this time, the critical region is established on the right side of the t-distribution in accordance with the t-distribution of the degree of freedom $(n - 1)$ as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \(H_0 : \mu = \mu_0\)
Alternative hypothesis \(H_1 : \mu > \mu_0\)

The test is performed using

\[
\frac{\bar{x} - \mu_0}{\frac{V}{\sqrt{n}}} > t(\alpha, n-1)
\]

\(\mu_0\) : population mean
\(V\) : unbiased variance
\(\bar{x}\) : sample mean
\(\alpha\) : significance level

\[V = \frac{\sum (x - \bar{x})^2}{n - 1}\]

**OPERATION**

6721 \[\text{UB}\]

Test \(H_0 : \mu = \mu_0\) \(H_1 : \mu > \mu_0\)
Input new data \((Y/N)\) ?-

The display appears as illustrated above once the library is activated. At this time, either \(Y\) or \(N\) should be pressed to perform the following procedures:

\(Y\) : New data input followed by test, additional data input, data edit, statistic check.
\(N\) : Test of previously stored data, test by inputting each value.

\((1)\) \(Y\)

\(Y\)

Input data \((x)\)

Input. Delete. Clear. List. End ?-

The menu display illustrated above appears when \(Y\) is pressed. One of the following character keys is then pressed to perform the corresponding function.

\(I\) (Input) : Data input (for input or addition of data).
\(D\) (Delete) : Data delete (for deletion of erroneous or unnecessary data).
\(C\) (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
	\( \text{ } \times \text{ or } \pm \text{ } \) scrolls to the following data item, \( \square \) to the previous item, and \( \Rightarrow \) or \( \square \) exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

\[
\begin{array}{c}
\text{N} \\
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\rho_0 = 0 ? -
\end{array}
\]

(Test display)

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

A company has conducted a survey of automobile expenses over a 5-month period. A previous 1-month survey revealed expenditures of $54.3. Using these results, determine whether or not expenditures have risen, with a significance level of 5%.

<table>
<thead>
<tr>
<th></th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXPENDITURES</td>
<td>72.4</td>
<td>62.3</td>
<td>58.4</td>
<td>55.4</td>
<td>64.8</td>
</tr>
</tbody>
</table>

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\text{input new data (Y/N)? -}
\end{array}
\]

(Select new data input.)

\[
\begin{array}{c}
\text{Y} \\
\text{input data (x) -}
\end{array}
\]

(Select data clear.)

\[
\begin{array}{c}
\text{C} \\
\text{clear data (Y/N)? -}
\end{array}
\]

(Data cleared.)

\[
\begin{array}{c}
\text{Y} \\
\text{input data (x) -}
\end{array}
\]

(Select data input.)

\[
\begin{array}{c}
\text{I} \\
\text{input data (x) [EXE]: menu -}
\end{array}
\]

(Enter first data item.)

\[
\begin{array}{c}
72.4 \text{ EX} \\
62.3 \text{ EX} \\
58.4 \text{ EX} \\
55.4 \text{ EX} \\
64.8 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{input data (x) [EXE]: menu -}
\end{array}
\]

(Enter remaining data items.)

\[
\begin{array}{c}
\text{E} \\
\text{return to menu. -}
\end{array}
\]

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\text{input data (x) -}
\end{array}
\]

(Enter population mean.)

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\text{input data (x) -}
\end{array}
\]

(Press EX after checking number of data.)

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\text{V= 42.608 ? -}
\end{array}
\]

(Press EX after checking data mean.)

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Significance level } \alpha(\%) \\
\text{z= 5 ? -}
\end{array}
\]

(Press EX after checking unbiased variance.)

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
2.864 > 2.132 : \text{ reject}
\end{array}
\]

(Enter significance level. 5% is already set, so simply press EX.)

\[
\begin{array}{c}
\text{E} \\
54.3 \text{ EX}
\end{array}
\]

\[
\begin{array}{c}
\text{Test } \mu=\mu_0 \quad H_1: \mu>\mu_0 \\
\mu_0 = 54.3 ? -
\end{array}
\]

(Display test result.)
Here, it can be said that automobile expenses have increased.

POPULATION MEAN TEST FLOWCHART (RIGHT SIDED FOR UNKNOWN VARIANCE)
Perform hypothesis testing of \( \mu \) in normal distribution \( N(\mu, \sigma^2; \mu: \text{unknown}, \sigma^2: \text{unknown}) \).

**CALCULATIONS**

An \( n \)-size sample \( (x_1, x_2, \ldots, x_n) \) is taken from normal distribution \( N(\mu, \sigma^2) \). At this time, the critical region is established on the left side of the \( t \)-distribution in accordance with the \( t \)-distribution of the degree of freedom \( (n-1) \) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0 : \mu = \mu_0 \)
Alternative hypothesis \( H_1 : \mu < \mu_0 \)

The test is performed using

\[
\frac{\bar{x} - \mu_0}{\sqrt{\frac{V}{n}}} < -t(c, n-1)
\]

\( t \)-distribution of degree of freedom \( (n-1) \)

\( \mu_c \) : population mean
\( V \) : unbiased variance
\( \bar{x} \) : sample mean
\( \alpha \) : significance level

\[
V = \frac{\sum (x - \bar{x})^2}{n-1}
\]

**OPERATION**

6722

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by test, additional data input, data edit, statistic check.
N : Test of previously stored data, test by inputting each value.

1. Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).

( or ) scrolls to the following data item, ( or ) to the previous item, and ( or ) exits the statistic display and returns to the menu.

E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

The table below shows the number of requests for after service of a product at a company which recently has changed its after service procedures. Under the old system, an average of 23 requests were received per month. Use the data to determine whether the new after service system has resulted in an improvement with a significance level of 1%.

<table>
<thead>
<tr>
<th></th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQUESTS</td>
<td>16</td>
<td>14</td>
<td>11</td>
<td>12</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test</th>
<th>Ho: μ = μ₀, H₁: μ &lt; μ₀</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input new data (Y/N) ? -</td>
<td></td>
</tr>
</tbody>
</table>

Y

Input data (x)
> Input. Delete. Clear. List. End ? -

C

Input data (x)
clear data (Y/N) ? -

Y

Input data (x)
> Input. Delete. Clear. List. End ? -

I

Input data (x) [EXE] : menu
x ? -

16 [EXE]

Input data (x) [EXE] : menu
x ? -

14 [EXE] 11 12 [EXE] 7 [EXE]

Input data (x) [EXE] : menu
x ? -

E

Input data (x)
> Input. Delete. Clear. List. End ? -

Test
μ₀ = 0 ? -

Test n = 5 ? -

Test T = 12 ? -

Test V = 11 5 ? -

Significance level α(%) = 5 ? -

Test Ho: μ = μ₀, H₁: μ < μ₀

Test -7 253 < 3 747 : Reject

Test μ₀ = 23 ? -

Here, it can be said that the number of requests for after service has decreased under the new system.
6730 
POPULATION VARIANCE TEST
(TWO-SIDED)

Performs hypothesis testing of $\sigma^2$ in normal distribution $N(\mu, \sigma^2)$; where $\mu$ : unknown, $\sigma^2$ : unknown.

CALCULATIONS

An $n$-size sample $(x_1, x_2 \ldots x_n)$ is taken from normal distribution $N(\mu, \sigma^2)$. At this time, critical regions are established on both sides of the $x^2$ distribution in accordance with the $x^2$-distribution of the degree of freedom $(n-1)$ as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) $H_0 : \sigma^2 = \sigma_0^2$
Alternative hypothesis $H_1 : \sigma^2 \neq \sigma_0^2$

The test is performed using
\[
\frac{S}{\sigma_0^2} < x^2(1 - \frac{\alpha}{2}, n - 1) \text{ or } \frac{S}{\sigma_0^2} > x^2(\frac{\alpha}{2}, n - 1)
\]

\[S = \sum (x - \bar{x})^2\]

OPERATION

6730 [LB]

Test
$H_0 : \sigma^2 = \sigma_0^2$
$H_1 : \sigma^2 \neq \sigma_0^2$
Input new data [Y/N] ?

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by test, additional data input, data edit, statistic check.
N : Test of previously stored data, test by inputting each value.

1) Y

Input data (x)
>Input, Delete, Clear, List, End ?

The menu display illustrated above appears when [Y] is pressed. One of the following character keys is then pressed to perform the corresponding function.

1 (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).

X (or EX) scrolls to the following data item, Q to the previous item, and Q or (or Q) exits the statistic display and returns to the menu.

E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

EXAMPLE

The following data represent the entrance examination results of five students. To date, the variance of scores for this test has been 70. Use the data to determine whether or not the variance of this year's scores is equivalent to past scores with a significance level of 1%.

<table>
<thead>
<tr>
<th>POINTS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>183</td>
<td>174</td>
<td>191</td>
<td>168</td>
<td>171</td>
</tr>
</tbody>
</table>

Test Ho : \( \sigma^2 = \sigma_0^2 \) H1 : \( \sigma^2 \neq \sigma_0^2 \)

Input new data (Y/N)?

Y

Input data (x)

C

Input data (x)

Y

Input data (x)

I

Input data (x) [EXE]: menu

183 EX

Input data (x) [EXE]: menu

174 EX 191 EX 168 EX 171 EX

Input data (x) [EXE]: menu

Enter remaining data items.

EX

Input data (x)

C

Input data (x)

E

Test Ho : \( \sigma^2 = \sigma_0^2 \) H1 : \( \sigma^2 \neq \sigma_0^2 \)

Test \( n = 5 \) ?

Test \( s = 357.2 \) ?

Significance level \( \alpha(\%) \)

\( \alpha = 5 \) ?

Test Ho : \( \sigma^2 = \sigma_0^2 \) H1 : \( \sigma^2 \neq \sigma_0^2 \)

Display test result.

Here, the variance of this year's scores is equivalent to last year's scores.
POPULATION VARIANCE TEST FLOWCHART (TWO-SIDED)

Library start

New data input

Menu 1

Y

Process selection

Menu 2

End

Data input

exit only?

Yes

No

Input of data to be deleted

exit only?

Yes

No

Data delete

Data input

Population variance display and input

(a²)

Number of data display and input

(n)

Sum of squares display and input

(S)

Significance level display and input

(α)

Test result display

(×)

Statistic display

(×)

Statistic present?

Yes

No

Next statistic display

Previous statistic display

Next statistic display
Population Variance Test (Right Sided)

Performs hypothesis testing of \( \sigma^2 \) in normal distribution N (\( \mu, \sigma^2 \); where \( \mu \) : unknown, \( \sigma^2 \) : unknown).

Calculations

An n-size sample (\( x_1, x_2, \ldots, x_n \)) is taken from normal distribution N (\( \mu, \sigma^2 \)). At this time, a critical region is established on the right side of the \( x^2 \) distribution in accordance with the \( x^2 \)-distribution of the degree of freedom (n - 1) as shown in the illustration when:

\[
\text{Hypothesis to be tested (Null hypothesis)} \quad H_0 : \sigma^2 = \sigma_0^2 \\
\text{Alternative hypothesis} \quad H_1 : \sigma^2 > \sigma_0^2
\]

The test is performed using

\[
\frac{S}{\sigma_0^2} > x^2(\alpha, n-1)
\]

\( x^2 \) distribution of degree of freedom (n - 1)  
\( \sigma^2 \) : population variance  
\( S \) : sum of squares  
\( \alpha \) : significance level  
\( S = \sum (x - \bar{x})^2 \)

Operation

6731 LB

Test  
H0 : \( \sigma^2 = \sigma_0^2 \)  
H1 : \( \sigma^2 > \sigma_0^2 \)  
Input new data [Y/N] ?

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y: New data input followed by test, additional data input, data edit, statistic check.  
N: Test of previously stored data, test by inputting each value.

(1) Y

Y

Input data (x)  
>Input, Delete, Clear, List, End ?

The menu display illustrated above appears when [Y] is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).  
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).  
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).

\( \text{[X]} \) or \( \text{[ES]} \) scrolls to the following data item, \( \text{[Q]} \) to the previous item, and \( \text{[C]} \) or \( \text{[ES]} \) exits the statistic display and returns to the menu.

E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

\[
\begin{array}{c}
\text{Test} \\
\sigma^2 = 0 \ ?_-
\end{array}
\]

(Test display)

The display appears as illustrated above when the \( \text{[N]} \) key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

A company purchased blades from Company B because the cost of the blades was less than those purchased from their usual supplier, Company A. The data in the table below represent the measured lengths of items cut with the Company B blades. To date, the variance of lengths of items cut with Company A blades has been 0.015. Use the data to compare the performance of the two companies' blades with a significance level of 1%.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>LENGTH</td>
<td>10.1</td>
<td>9.8</td>
<td>10.2</td>
<td>9.7</td>
<td>9.8</td>
</tr>
</tbody>
</table>

\[
\begin{array}{c}
\text{Test} \\
\sigma^2 = 0 \ ?_-
\end{array}
\]

(Select new data input.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
> \text{Input.Delete.Clear.List.End} \ ?_-
\end{array}
\]

(Select data clear.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
> \text{Clear data} \ (Y/N) \ ?
\end{array}
\]

(Data cleared.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
> \text{Input.Delete.Clear.List.End} \ ?_-
\end{array}
\]

(Select data input.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
\text{[EXE]} : \text{menu} \ x?_-
\end{array}
\]

(Enter first data item.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
\text{[EXE]} : \text{menu} \ x?_-
\end{array}
\]

(Enter remaining data items.)

\[
\begin{array}{c}
\text{Input data} \ (x) \\
> \text{Input.Delete.Clear.List.End} \ ?_-
\end{array}
\]

(Return to menu.)

\[
\begin{array}{c}
\text{Test} \\
\sigma^2 = 0 \ ?_-
\end{array}
\]

(Select End.)

\[
\begin{array}{c}
\text{Test} \\
n = 5 \ ?_-
\end{array}
\]

(Enter population variance.)

\[
\begin{array}{c}
\text{Test} \\
S^2 = 0.188 \ ?_-
\end{array}
\]

(Press \( \text{[CH]} \) after checking number of data.)

\[
\begin{array}{c}
\text{Significance level \( \alpha \)} \% \\
e = 5 \ ?_-
\end{array}
\]

(Enter checking significance level.)

\[
\begin{array}{c}
\text{Test} \\
\text{12.53} \leq 13.28 \ : \text{Accept}
\end{array}
\]

(Enter significance level.)

\[
\begin{array}{c}
\text{Test} \\
\sigma^2 = 0.015 \ ?_-
\end{array}
\]

(Enter test result.)

Here, it is determined that the performance of both companies' blades are equivalent.
6732 POPULATION VARIANCE TEST
(LEFT SIDED)

Performs hypothesis testing of \( \sigma^2 \) in normal distribution \( N(\mu, \sigma^2) \); where \( \mu \) : unknown, \( \sigma^2 \) : unknown.

CALCULATIONS
An \( n \)-size sample \( (x_1, x_2 \ldots x_n) \) is taken from normal distribution \( N(\mu, \sigma^2) \). At this time, a critical region is established on the left side of the \( \chi^2 \) distribution in accordance with the \( \chi^2 \) distribution of the degree of freedom \( n-1 \) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0 : \sigma^2 = \sigma_0^2 \)
Alternative hypothesis \( H_1 : \sigma^2 < \sigma_0^2 \)

The test is performed using

\[
\frac{S}{\sigma_0^2} < \chi^2(1-\alpha,n-1)
\]

\( \chi^2 \) distribution of degree of freedom \( n-1 \)

\[
\begin{align*}
\chi^2(1-\alpha,n-1) & : \text{population variance} \\
S & : \text{sum of squares} \\
\alpha & : \text{significance level} \\
S & = \sum (x-x)^2
\end{align*}
\]

OPERATION

6732 LB

Table

\[
\begin{array}{ccc}
H_0 : \sigma^2 = \sigma_0^2 & H_1 : \sigma^2 < \sigma_0^2 \\
\text{input new data (Y/N) ?} & \\
\end{array}
\]

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by test, additional data input, data edit, statistic check.
N : Test of previously stored data, test by inputting each value.

(1) Y

\[
\begin{array}{c}
\text{input data (x)} \\
> \text{input, delete, clear, list, end ?}
\end{array}
\]

The menu display illustrated above appears when (Y) is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
U or D scrolls to the following data item, L to the previous item, and E or C exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

Test $H_0: \mu = \mu_0$  $H_1: \mu < \mu_0$

The display appears as illustrated above when the N key is pressed. From this point, various parameters are entered for the test.

**EXAMPLE**

A company has purchased a new production machinery. The data in the table below represent the production capacity of the new machinery. To date, the variance of production capacity for the old machinery has been 0.1. Use the data to compare the performance of the machinery with a significance level of 1%.

<table>
<thead>
<tr>
<th>WEIGHT (g)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>70.0</td>
<td>69.9</td>
<td>70.1</td>
<td>70.1</td>
<td>69.8</td>
</tr>
</tbody>
</table>

Test $H_0: \mu = \mu_0$  $H_1: \mu < \mu_0$

Input new data: (Y/N)?

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

[EXE] MENU

70.0 D

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

[EXE] MENU

69.9 D 70.1 D 70.1 D 69.8 D

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

Input data [x] or Input Delete Clear List End?

[EXE] MENU

[EXE] MENU

[EXE] MENU

[EXE] MENU

[EXE] MENU

Here, it is determined that the performance of the new machinery is equivalent to that of the old machinery.
VARIANCE RATIO TEST
(TWO-SIDED)

Performs test of hypotheses \( \sigma_1^2 \) and \( \sigma_2^2 \) in two normal distributions \( N(\mu_1, \sigma_1^2) \) and \( N(\mu_2, \sigma_2^2) \) where \( \mu_1 \) : unknown, \( \sigma_1^2 \) : unknown) and \( N(\mu_2, \sigma_2^2) \) where \( \mu_2 \) : unknown, \( \sigma_2^2 \) : unknown).

CALCULATIONS

An \( n_1 \)-size sample \((x_{11}, x_{12}, \ldots, x_{1n_1})\) is taken from normal distribution \( N(\mu_1, \sigma_1^2) \) and an \( n_2 \)-sample \((x_{21}, x_{22}, \ldots, x_{2n_2})\) from normal distribution \( N(\mu_2, \sigma_2^2) \). At this time, critical regions are established on both sides of the F distribution in accordance with the F distribution of the degrees of freedom \((n_1 - 1, n_2 - 1)\) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0 : \sigma_1^2 = \sigma_2^2 \)
Alternative hypothesis \( H_1 : \sigma_1^2 \neq \sigma_2^2 \)

The test is performed using

\[
V_1 < V_2 : \quad \frac{V_2}{V_1} > F \left( \frac{\sigma_1^2}{\sigma_2^2}, n_2 - 1, n_1 - 1 \right)
\]

\[
V_1 > V_2 : \quad \frac{V_1}{V_2} > F \left( \frac{\sigma_2^2}{\sigma_1^2}, n_1 - 1, n_2 - 1 \right)
\]

\[
F = \frac{\sum (x - \bar{x})^2}{n - 1}
\]

OPERATION

6740

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by test, additional data input, data edit, statistic check.
N : Test of previously stored data, test by inputting each value.

(1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
( or ) scrolls to the following data item, to the previous item, and exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items through x2n, while the data input being queried on the original display is for data items x1 through x1n.

(2-1) Y
Same result as that produced by pressing Y in step (1) above. Note, however, that the data being entered or corrected here is x2 through x2n.

(2-2) N

The display appears as illustrated above when the N key is pressed. The value indicated for n shows the number of x1 (x1 through x1n) data currently stored in memory.

- n1 = 0 : Test cannot be performed, so this should be corrected to the required data.
- Number of data input (following Y above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
- Number of data input (following Y above) matches value of n : Press N.
When N is pressed, a display similar to that above is produced for x2 (x2 through x2n) data items. After confirmation and/or corrections as described in (2-2), press N to continue.

EXAMPLE
The following data represent measurement results on samples taken from two lines in a factory. Use the data to determine whether or not production on the two lines differ with a significance level of 5%.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE A</td>
<td>37.2</td>
<td>38.1</td>
<td>39.9</td>
<td>37.5</td>
<td>36.1</td>
</tr>
<tr>
<td>LINE B</td>
<td>36.1</td>
<td>35.2</td>
<td>37.7</td>
<td>35.6</td>
<td>—</td>
</tr>
</tbody>
</table>

N

(Select new data input.)
Here, it is determined that the variance for the output of both lines are equivalent.
VARIANCE RATIO TEST
(RIGHT SIDED)

Performs hypotheses testing of \( \sigma_1^2 \) and \( \sigma_2^2 \) in two normal distributions \( N (\mu_1, \sigma_1^2) \) and \( N (\mu_2, \sigma_2^2) \) where \( \mu_1 \), \( \sigma_1^2 \), \( \mu_2 \), and \( \sigma_2^2 \) are unknown.

CALCULATIONS

An \( n_1 \)-size sample \( (x_{11}, x_{12}, \ldots, x_{1n_1}) \) is taken from normal distribution \( N (\mu_1, \sigma_1^2) \) and an \( n_2 \) sample \( (x_{21}, x_{22}, \ldots, x_{2n_2}) \) from normal distribution \( N (\mu_2, \sigma_2^2) \). At this time, the critical region is established on the right side of the \( F \) distribution in accordance with the \( F \) distribution of the degrees of freedom \( (n_1 - 1, n_2 - 1) \) as shown in the illustration when:

- Hypothesis to be tested (Null hypothesis) \( H_0: \sigma_1^2 = \sigma_2^2 \)
- Alternative hypothesis \( H_1: \sigma_1^2 > \sigma_2^2 \)

The test is performed using

\[
\frac{\sigma_1^2}{\sigma_2^2} \sim F (n_1 - 1, n_2 - 1)
\]

\( F \) distribution of degrees of freedom \( (n_1 - 1, n_2 - 1) \)

\[ V = \frac{\sum (x - \bar{x})^2}{n - 1} \]

OPERATION

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y: New data input followed by test, additional data input, data edit, statistic check.

N: Test of previously stored data, test by inputting each value.

(1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
(List) Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).

or scrolls to the following data item, to the previous item, and or exits the statistic display and returns to the menu.

(End) Advances to the test display (same as when N is pressed in the first step above).

!) N

Input new data \( x \) \((Y/N) ?\)

The display appears as illustrated above when the key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items \( x_2 \) through \( x_{2n} \), while the data input being queried on the original display is for data items \( x_1 \) through \( x_n \).

2-1) Y

Same result as that produced by pressing \( Y \) in step (1) above. Note, however, that the data being entered or corrected here is \( x_2 \) through \( x_{2n} \).

2-2) N

(Number of data display)

The display appears as illustrated above when the key is pressed. The value indicated for \( n \) shows the number of \( x_1 \) (\( x_{11} - x_{1n} \)) data currently stored in memory.

\( n = 0 \): Test cannot be performed, so the required data should be corrected to the required data.

Number of data input (following \( Y \) above) and value of \( n \) differ: Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.

Number of data input (following \( X \) above) matches value of \( n \): Press \( X \).

When \( X \) is pressed, a display similar to that above is produced for \( x_2 \) (\( x_{21} - x_{2n} \)) data items. After confirmation and/or corrections as described in (2-2), press \( X \) to continue.

**Example**
The following data represent the number of customers at a restaurant before and after recent remodeling. Use the data to determine whether or not the number of customers has been stabilized by the renovation with a significance level of 5%.

<table>
<thead>
<tr>
<th>NUMBER OF CUSTOMERS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEFORE REMODELING</td>
<td>114</td>
<td>120</td>
<td>78</td>
<td>151</td>
<td>63</td>
</tr>
<tr>
<td>AFTER REMODELING</td>
<td>127</td>
<td>120</td>
<td>138</td>
<td>141</td>
<td>—</td>
</tr>
</tbody>
</table>
Here, it is determined that the number of customers has not been affected by the remodeling.
VARIANCE RATIO TEST
(LEFT SIDED)

Performs hypotheses testing of \( \sigma_1^2 \) and \( \sigma_2^2 \) in two normal distributions \( N(\mu_1, \sigma_1^2) \) where \( \mu_1 \) unknown, \( \sigma_1^2 \) unknown and \( N(\mu_2, \sigma_2^2) \) where \( \mu_2 \) unknown, \( \sigma_2^2 \) unknown.

CALCULATIONS
An \( n_1 \)-size sample \( (x_{11}, x_{12}, \ldots, x_{1n_1}) \) is taken from normal distribution \( N(\mu_1, \sigma_1^2) \) and an \( n_2 \) sample \( (x_{21}, x_{22}, \ldots, x_{2n_2}) \) from normal distribution \( N(\mu_2, \sigma_2^2) \). At this time, the critical region is established on the right side of the F distribution in accordance with the F distribution of the degrees of freedom \( (n_1-1, n_2-1) \) as shown in the illustration when:
Hypothesis to be tested (Null hypothesis) \( H_0 : \sigma_1^2 = \sigma_2^2 \)
Alternative hypothesis \( H_1 : \sigma_1^2 < \sigma_2^2 \)
The test is performed using
\[
\frac{V_1}{V_2} < F(1-\alpha, n_1-1, n_2-1) \text{ or } \frac{V_2}{V_1} > F(\alpha, n_2-1, n_1-1)
\]

\[ F \text{ distribution of degrees of freedom } (n_1-1, n_2-1) \]

\[ V = \sum \frac{(x-x)^2}{n-1} \]

OPERATION

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y: New data input followed by test, additional data input, data edit, statistic check.
N: Test of previously stored data, test by inputting each value.

(1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.

I (Input): Data input (for input or addition of data).
D (Delete): Data delete (for deletion of erroneous or unnecessary data).
C (Clear): Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
( or ) scrolls to the following data item, ( or ) to the previous item, and ( or ) exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the (N) key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items x2, through x2m, while the data input being queried on the original display is for data items x1, through x1n.

(2-1) Y
Same result as that produced by pressing (Y) in step (1) above. Note, however, that the data being entered or corrected here is x2, through x2m.

(2-2) N

The display appears as illustrated above when the (N) key is pressed. The value indicated for n shows the number of x1 (x1, x1n) data currently stored in memory.

• n 0 : Test cannot be performed, so this should be corrected to the required data.
• Number of data input (following (Y) above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
• Number of data input (following (Y) above) matches value of n : Press (X). When (X) is pressed, a display similar to that above is produced for x2 (x2, x2m) data items. After confirmation and/or corrections as described in (2-2), press (X) to continue.

EXAMPLE
The following data represent the number of customers at a store before and after a recent change in the main line of products. Use the data to determine whether or not the number of customers has decreased since the change with a significance level of 5%.

<table>
<thead>
<tr>
<th>NUMBER OF CUSTOMERS</th>
<th>PRODUCT A</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT B</td>
<td>251</td>
<td>238</td>
<td>261</td>
<td>220</td>
<td>243</td>
<td></td>
</tr>
<tr>
<td></td>
<td>241</td>
<td>268</td>
<td>224</td>
<td>230</td>
<td>—or</td>
<td></td>
</tr>
</tbody>
</table>

Test : \( H_0 : \sigma_1^2 = \sigma_2^2 \), \( H_1 : \sigma_1^2 < \sigma_2^2 \)
Input new data x2 (Y/N) ?

Input data (x1) >Input Delete Clear List End ?

(Select new data input)
Here, it is determined that the number of customers has remained the same since the product change.
Performs hypotheses testing of \( \mu_1 \) and \( \mu_2 \) in two normal distributions \( N(\mu_1, \sigma^2) \) where \( \mu_1 \) unknown, \( \sigma^2 \) unknown) and \( N(\mu_2, \sigma^2) \) where \( \mu_2 \) unknown, \( \sigma^2 \) unknown.

**CALCULATIONS**

An \( n_1 \)-size sample \((x_{11}, \ldots, x_{1n_1})\) is taken from normal distribution \( N(\mu_1, \sigma^2) \) and an \( n_2 \) sample \((x_{21}, x_{22}, \ldots, x_{2n_2})\) from normal distribution \( N(\mu_2, \sigma^2) \). At this time, critical regions are established on both sides of the t-distribution in accordance with the t-distribution of the degree of freedom \((n_1 + n_2 - 2)\) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0: \mu_1 = \mu_2 \)

Alternative hypothesis \( H_1: \mu_1 \neq \mu_2 \)

The test is performed using

\[
\left| \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \right| > t\left(\frac{\sigma}{2}, n_1 + n_2 - 2\right)
\]

\( \bar{x}_1 \): sample mean 1
\( \bar{x}_2 \): sample mean 2
\( S_1 \): sum of squares 1
\( S_2 \): sum of squares 2
\( \sigma^2 \): population variance
\( \alpha \): significance level

\( S = \sum (x - \bar{x})^2 \)

**OPERATION**

6750

Test \( H_0: \mu_1 = \mu_2 \) \( H_1: \mu_1 \neq \mu_2 \)

Input new data \((x_1, \ldots, x_n)\) \((Y/N)\)

The display appears as illustrated above once the library is activated. At this time, either \( Y \) or \( N \) should be pressed to perform the following procedures:

\( Y \): New data input followed by test, additional data input, data edit, statistic check.

\( N \): Test of previously stored data, test by inputting each value.

(1) \( Y \)

\( \triangleright \) Input data \((x_1)\)

The menu display illustrated above appears when \( Y \) is pressed. One of the following character keys is then pressed to perform the corresponding function.

<table>
<thead>
<tr>
<th>I (Input)</th>
<th>Data input (for input or addition of data).</th>
</tr>
</thead>
<tbody>
<tr>
<td>D (Delete)</td>
<td>Data delete (for deletion of erroneous or unnecessary data).</td>
</tr>
<tr>
<td>C (Clear)</td>
<td>Data clear (for deletion of previously stored data. This operation also clears statistics).</td>
</tr>
</tbody>
</table>
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation). 
( or ) scrolls to the following data item, ( or ) to the previous item, and ( or ) exits the statistic display and returns to the menu.

E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

Test Ho : \( \mu_1 = \mu_2 \) H1 : \( \mu_1 \neq \mu_2 \)
Input new data x? (Y/N)?

The display appears as illustrated above when the (N) key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items \( x_2 \) through \( x_{2n} \), while the data input being queried on the original display is for data items \( x_1 \) through \( x_{1n} \).

(2-1) Y
Same result as that produced by pressing (Y) in step (1) above. Note, however, that the data being entered or corrected here is \( x_2 \) through \( x_{2n} \).

(2-2) N

Test
\[ n_1 = 5 \]
Ho : \( \mu_1 = \mu_2 \) H1 : \( \mu_1 \neq \mu_2 \) (Number of data display)

The display appears as illustrated above when the (N) key is pressed. The value indicated for \( n \) shows the number of \( x_1 \) (\( x_{11} \sim x_{1n} \)) data currently stored in memory.

* \( n_1 = 0 \) : Test cannot be performed, so this should be corrected to the required data.
* Number of data input (following (Y) above) and value of \( n \) differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
* Number of data input (following (Y) above) matches value of \( n \) : Press (N).
When (N) is pressed, a display similar to that above is produced for \( x_2 \) (\( x_{21} \sim x_{2n} \)) data items. After confirmation and/or corrections as described in (2-2), press (N) to continue.

**EXAMPLE**
The following data represent the results of durability tests on ten products, five each from two different factories. Use the data to determine whether or not the quality of the products manufactured at the factories differ with a significance level of 5%.

<table>
<thead>
<tr>
<th>DURABILITY (HOURS)</th>
<th>FACTORY A</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FACTORY B</td>
<td>853</td>
<td>844</td>
<td>850</td>
<td>854</td>
<td>844</td>
</tr>
</tbody>
</table>

Test
Ho : \( \mu_1 = \mu_2 \) H1 : \( \mu_1 \neq \mu_2 \)
Input new data x1 (Y/N)?

(Select new data input)
Here, it is determined that the quality of goods manufactures at the two factories is equivalent.
MEAN DIFFERENCE TEST (RIGHT SIDED)

Performs hypotheses testing of \( \mu_1 \) and \( \mu_2 \) in two normal distributions \( N(\mu_1, \sigma^2) \) where \( \mu_1 \): unknown, \( \sigma^2 \): unknown) and \( N(\mu_2, \sigma^2) \) where \( \mu_2 \): unknown \( \sigma^2 \): unknown)

**CALCULATIONS**

An \( n_1 \)-size sample \( (x_{11}, x_{12}, \ldots, x_{1n_1}) \) is taken from normal distribution \( N(\mu_1, \sigma^2) \) and an \( n_2 \) sample \( (x_{21}, x_{22}, \ldots, x_{2n_2}) \) from normal distribution \( N(\mu_2, \sigma^2) \). At this time, the critical region is established on the right of the t-distribution in accordance with the t-distribution of the degree of freedom \( (n_1 + n_2 - 2) \) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0 : \mu_1 = \mu_2 \)
Alternative hypothesis \( H_1 : \mu_1 > \mu_2 \)

The test is performed using

\[
\frac{x_1 - x_2}{\sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \left( \frac{S_1^2 + S_2^2}{n_1 + n_2 - 2} \right)^{\frac{1}{2}} } > t(\alpha, n_1 + n_2 - 2)
\]

**OPERATION**

The display appears as illustrated above once the library is activated. At this time, either Y or N should be pressed to perform the following procedures:

Y : New data input followed by test, additional data input, data edit, statistic check.
N : Test of previously stored data, test by inputting each value.

(1) Y

The menu display illustrated above appears when Y is pressed. One of the following character keys is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
    (or ) scrolls to the following data item, to the previous item, and exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the N key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items x₁ through x₂n, while the data input being queried on the original display is for data items x₁₁ through x₁m.

(2-1) Y
Same result as that produced by pressing Y in step (1) above. Note, however, that the data being entered or corrected here is x₂₁ through x₂n₂.

(2-2) N

The display appears as illustrated above when the N key is pressed. The value indicated for n shows the number of x₁ (x₁₁ ~ x₁m) data currently stored in memory.

- n₁ = 0 : Test cannot be performed, so this should be corrected to the required data.
- Number of data input (following Y above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
- Number of data input (following Y above) matches value of n : Press EM.
When is pressed, a display similar to that above is produced for x₂ (x₂₁ ~ x₂n₂) data items. After confirmation and/or corrections as described in (2-2), press EM to continue.

**EXAMPLE**
The following data represent the results of tests on light bulb A which is more expensive than light bulb B, but claims a longer service life. Use the data to determine whether or not there is a difference in the service lives of light bulbs A and B with a significance level of 5%.

<table>
<thead>
<tr>
<th>TIME</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>890</td>
<td>880</td>
<td>920</td>
<td>870</td>
<td>900</td>
</tr>
<tr>
<td>B</td>
<td>850</td>
<td>840</td>
<td>870</td>
<td>855</td>
<td>860</td>
</tr>
</tbody>
</table>

---

379
Here, it is determined that the service life of LIGHT BULB A is longer than that of LIGHT BULB B.
MEAN DIFFERENCE TEST (LEFT SIDED)

Performs hypotheses testing of $\mu_1$ and $\mu_2$ in two normal distributions $N(\mu_1, \sigma^2)$; where $\mu_1$: unknown, $\sigma^2$: unknown) and $N(\mu_2, \sigma^2)$; where $\mu_2$: unknown, $\sigma^2$: unknown)

CALCULATIONS
An $n_1$-size sample ($x_{11}, x_{12}, \ldots, x_{n_1}$) is taken from normal distribution $N(\mu_1, \sigma^2)$ and an $n_2$ sample ($x_{21}, x_{22}, \ldots, x_{n_2}$) from normal distribution $N(\mu_2, \sigma^2)$. At this time, the critical region is established on the left of the t-distribution in accordance with the t-distribution of the degree of freedom $(n_1 + n_2 - 2)$ as shown in the illustration when:
Hypothesis to be tested (Null hypothesis) $H_0: \mu_1 = \mu_2$
Alternative hypothesis $H_1: \mu_1 < \mu_2$

The test is performed using

$$
\frac{x_1 - x_2}{\sqrt{\left(\frac{1}{n_1} + \frac{1}{n_2}\right)\left(\frac{S_1 + S_2}{n_1 + n_2 - 2}\right)}} < -t(\alpha; n_1 + n_2 - 2)
$$

OPERATION

6752

The display appears as illustrated above once the library is activated. At this time, either $Y$ or $N$ should be pressed to perform the following procedures:

$Y$: New data input followed by test, additional data input, data edit, statistic check.
$N$: Test of previously stored data, test by inputting each value.

(1) $Y$

$\check{Y}$

The menu display illustrated above appears when $\check{Y}$ is pressed. One of the following character keys is then pressed to perform the corresponding function.
I (Input) : Data input (for input or addition of data).
D (Delete) : Data delete (for deletion of erroneous or unnecessary data).
C (Clear) : Data clear (for deletion of previously stored data. This operation also clears statistics).
L (List) : Statistic display (for display of number of data items, sum, sum of squares, mean, population standard deviation, sample standard deviation).
[ or ] scrolls to the following data item, [ to the previous item, and [ or ] exits the statistic display and returns to the menu.
E (End) : Advances to the test display (same as when N is pressed in the first step above).

(2) N

The display appears as illustrated above when the [N] key is pressed. Note that this display is almost identical to the initial display which appears immediately after entering library operations. The difference, however, is that the question concerning new data input here is for data items x2 through x2n, while the data input being queried on the original display is for data items x1 through x1n.

(2-1) Y

Same result as that produced by pressing [Y] in step (1) above. Note, however, that the data being entered or corrected here is x2 through x2n.

(2-2) N

The display appears as illustrated above when the [N] key is pressed. The value indicated for n shows the number of x1 (x11 ~ x1n) data currently stored in memory.

- n1 = 0 : Test cannot be performed, so this should be corrected to the required data.
- Number of data input (following [Y] above) and value of n differ : Confirm that some data have not been omitted during the input or that two or more data items have been input together for a single entry. In either case, terminate the library operation. Enter the library again and add, delete, or reinput data as required.
- Number of data input (following [Y] above) matches value of n : Press [EX].
  When [EX] is pressed, a display similar to that above is produced for x2 (x21 ~ x2n) data items. After confirmation and/or corrections as described in (2-2), press [EX] to continue.

**EXAMPLE**
The following data represent the results of tests on concrete samples. SAMPLE A is not reinforced, while SAMPLE B is reinforced. Use the data to determine whether or not the reinforcement actually makes the concrete stronger with a significance level of 1%.

<table>
<thead>
<tr>
<th>STRENGTH (kg)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAMPLE A</td>
<td>18</td>
<td>20</td>
<td>17</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>SAMPLE B</td>
<td>25</td>
<td>24</td>
<td>22</td>
<td>26</td>
<td>24</td>
</tr>
</tbody>
</table>
Here, it is determined that the strength of SAMPLE B is greater than that of SAMPLE A.
RATIO TEST (RIGHT SIDED)

Performs test of hypothesis of population p ratio in binomial distribution B (1, p).

CALCULATIONS
An n-size sample (x1, x2, ..., xn) is taken from binomial distribution B (1, p). At this time, the critical region is established on the right side of the normal distribution in accordance with the approximate standard normal distribution N (0, 1^2) as shown in the illustration when:
Hypothesis to be tested (Null hypothesis)  H0: p = p0
Alternative hypothesis                  H1: p > p0

The test is performed using

\[
\frac{\sum x - np_0}{\sqrt{np_0(1-p_0)}} > \alpha(a)
\]

![Normal distribution curve]

(\(p_0\) : population ratio
\(\alpha\) : significance level

OPERATION

6761 [LB]

**Example**

A company has sent out 2,000 pieces of direct mail advertising printed in four colors and, as a result, received 80 orders. In the past, direct mail has produced a response of 2.5%. Determine whether or not the 4-color direct mailing was as effective as those in the past with a significance level of 5%.

<table>
<thead>
<tr>
<th>Test n = 0 ?</th>
<th>Ho: p = p0</th>
<th>H1: p &gt; p0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>Ho: p = p0</td>
<td>H1: p &gt; p0</td>
</tr>
<tr>
<td>3.065</td>
<td>0.025</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>(Enter number of data items)</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>(Enter number of responses)</td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>(Enter significance level, 5% is already set, so simply press [EX])</td>
<td></td>
</tr>
</tbody>
</table>

Test: Ho: p = p0 H1: p > p0

4.297 > 1.645 : Reject

Test: Ho: p = p0 H1: p > p0

p = 0.025 ?

(Enter probability)

(Display test result)

(Return to original display)

Here, it is determined that the 4-color mailing was more effective than those in the past.
RATIO TEST (LEFT SIDED)

Performs test of hypothesis of population p ratio in binomial distribution B (1, p).

CALCULATIONS
An n-size sample (x1, x2…xn) is taken from binomial distribution B (1, p). At this time, the critical region is established on the left side of the normal distribution in accordance with the approximate standard normal distribution N (0, 1²) as shown in the illustration when:
Hypothesis to be tested (Null hypothesis) Ho : p = p0
Alternative hypothesis H1 : p < p0

The test is performed using
\[ \frac{\sum x - np_0}{\sqrt{np_0(1 - p_0)}} < -\mu (\alpha) \]

Normal distribution curve
\( p_0 : \) population ratio
\( \alpha : \) significance level

OPERATION

EXAMPLE
A factory had a defect rate of 2.5% of products. After improvements in the process, a total of 18 defective products were detected in 1,000 items. Determine whether or not the improvements have caused the defect rate to decrease with a significance level of 1%.

0.025

1000

18

1

Here, it is determined that the improvements have not produced a decrease in the defect rate.
RATIO TEST (TWO-SIDED)

Performs test of hypothesis of population ratio \( p \) in binomial distribution B (1, \( p \)).

CALCULATIONS

An n-size sample \((x_1, x_2, \ldots, x_n)\) is taken from binomial distribution B (1, \( p \)). At this time, critical regions are established on both sides of the normal distribution in accordance with the approximate standard normal distribution \(N(0, 1^2)\) as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) \( H_0: p = p_0 \)
Alternative hypothesis \( H_1: p \neq p_0 \)

The test is performed using

\[
\left| \frac{\sum x - np_0}{\sqrt{np_0(1-p_0)}} \right| > \Phi \left( \frac{a}{2} \right)
\]

\( \Phi \) : population ratio
\( a \) : significance level

OPERATION

6760 [LB]

EXAMPLE

The crossbreeding of a certain type of bean should result in a ratio of 3:1 of yellow beans to green. An actual sample reveals 310 yellow beans within a total of 400. Determine whether or not this is equivalent to the 3:1 ratio noted above with a significance level of 5%.

(HINT: Test the hypothesis \( H : p = \frac{3}{4} \))

0.75 [EX]

400 [EX]

310 [EX]

[EX]

[EX]

Here, it is determined that the sample mixture is equivalent to the 3:1 ratio.
PERFORMS HYPOTHESES TESTING OF $p_1$ AND $p_2$ IN TWO BINOMIAL DISTRIBUTIONS B (1, $p_1$) AND B (1, $p_2$).

CALCULATIONS
An $n_1$-size sample $(x_{11}, x_{12}, \ldots, x_{1m_1})$ is taken from binomial distribution B (1, $p_1$) and an $n_2$ sample $(x_{21}, x_{22}, \ldots, x_{2m_2})$ from binomial distribution B (1, $p_2$). At this time, critical regions are established on both sides of the normal distribution in accordance with the approximate standard normal distribution $N(0, 1^2)$ as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) $H_0: p_1 = p_2$

Alternative hypothesis $H_1: p_1 \neq p_2$

The test is performed using

$$
\frac{\left| \frac{\sum x_{1i}}{n_1} - \frac{\sum x_{2i}}{n_2} \right|}{\sqrt{\frac{p(1-p)}{n_1} + \frac{(1-p)}{n_2}}} > u\left(\frac{\alpha}{2}\right)
$$

Where $p = \frac{\sum x_{1i} + \sum x_{2i}}{n_1 + n_2}$

Normal distribution curve (Sample ratio 1 $p_1$; Sample ratio 2 $p_2$; Significance level $\alpha$)

OPERATION

6770 LB

Test $n_1 \neq 0$ ?

$H_0: p_1 = p_2 \quad H_1: p_1 \neq p_2$

EXAMPLE

The following data represent the results of a survey taken for a certain product. Use the data to determine whether or not opinions differ according to gender with a significance level of 5%.

<table>
<thead>
<tr>
<th></th>
<th>LIKE</th>
<th>DISLIKE</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>MALE</td>
<td>260</td>
<td>140</td>
<td>400</td>
</tr>
<tr>
<td>FEMALE</td>
<td>180</td>
<td>120</td>
<td>300</td>
</tr>
</tbody>
</table>

400 LB

Test $n_1 \neq 0$ ?

$H_0: p_1 = p_2 \quad H_1: p_1 \neq p_2$

(Enter number of males)

260 LB

Test $n_1 \neq 0$ ?

$H_0: p_1 = p_2 \quad H_1: p_1 \neq p_2$

(Enter number of males answering LIKE)

300 LB

Test $n_1 \neq 0$ ?

$H_0: p_1 = p_2 \quad H_1: p_1 \neq p_2$

(Enter number of females)
Here, it is determined that there is no difference in the opinions of males and females.
RATIO DIFFERENCE TEST (RIGHT SIDED)

Performs hypotheses testing of \( p_1 \) and \( p_2 \) in two binomial distributions \( B(1, p_1) \) and \( B(1, p_2) \).

CALCULATIONS
An \( n_1 \)-size sample \((x_{11}, x_{12}, \ldots, x_{1n_1})\) is taken from binomial distribution \( B(1, p_1) \) and an \( n_2 \) sample \((x_{21}, x_{22}, \ldots, x_{2n_2})\) from binomial distribution \( B(1, p_2) \). At this time, a critical region is established on the right side of the normal distribution in accordance with the approximate standard normal distribution \( N(0, 1^2) \) as shown in the illustration when:

- Hypothesis to be tested (Null hypothesis) \( H_0 : p_1 = p_2 \)
- Alternative hypothesis \( H_1 : p_1 > p_2 \)

The test is performed using

\[
\sqrt{\frac{p(1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}{p + \frac{\sum x_1 + \sum x_2}{n_1 + n_2}}} > u(\alpha)
\]

\( u(\alpha) \) is the critical value of the normal distribution curve

\( p_1 \): sample ratio 1
\( p_2 \): sample ratio 2
\( \alpha \): significance level

OPERATION

EXAMPLE

The following data represent samples taken of the same product manufactured at two different factories. Use the data to determine whether the defect rate is greater for FACTORY A with a significance level of 5%.

<table>
<thead>
<tr>
<th>SAMPLES</th>
<th>DEFECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>FACTORY A</td>
<td>600</td>
</tr>
<tr>
<td>FACTORY B</td>
<td>400</td>
</tr>
</tbody>
</table>

\( n_1 = 600 \) K.H.

\( n_1 = 600 \) K.H.

(Enter number of samples from FACTORY A)
<table>
<thead>
<tr>
<th>Test</th>
<th>Ho: (d_t \leq d_e)</th>
<th>H1: (d_t &gt; d_e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>Ho: (d_t \leq d_e)</td>
<td>H1: (d_t &gt; d_e)</td>
</tr>
<tr>
<td>Test</td>
<td>Ho: (d_t \leq d_e)</td>
<td>H1: (d_t &gt; d_e)</td>
</tr>
<tr>
<td>Test</td>
<td>Ho: (d_t \leq d_e)</td>
<td>H1: (d_t &gt; d_e)</td>
</tr>
<tr>
<td>Test</td>
<td>Ho: (d_t \leq d_e)</td>
<td>H1: (d_t &gt; d_e)</td>
</tr>
</tbody>
</table>

(Enter number of defects)
(Enter number of samples from FACTORY B)
(Enter number of defects)
(Enter significance level. 5% is already set, so simply press \(\text{ENT}\))
(Display test result)
(Return to initial display)

Here, it is determined that there is no difference in the defect rate for the two factories.
Performs hypotheses testing of $p_1$ and $p_2$ in two binomial distributions $B(1, p_1)$ and $B(1, p_2)$.

**CALCULATIONS**

An $n_1$-size sample $(x_{11}, x_{12}, \ldots x_{1n_1})$ is taken from binomial distribution $B(1, p_1)$ and an $n_2$ sample $(x_{21}, x_{22}, \ldots x_{2n_2})$ from binomial distribution $B(1, p_2)$. At this time, a critical region is established on the left side of the normal distribution in accordance with the approximate standard normal distribution $N(0, 1^2)$ as shown in the illustration when:

Hypothesis to be tested (Null hypothesis) $H_0 : p_1 = p_2$

Alternative hypothesis $H_1 : p_1 < p_2$

The test is performed using

$$\frac{\sum x_{1} - \sum x_{2}}{\sqrt{n_1 p (1-p) \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} > u(\alpha)$$

Where

$p = \frac{\sum x_{1} + \sum x_{2}}{n_1 + n_2}$

Normal distribution curve

$p_1$ : sample ratio 1
$p_2$ : sample ratio 2
$\alpha$ : significance level

**OPERATION**

6772 UB

**EXAMPLE**

The following data represent the results of a survey taken in two areas concerning recognition of a product. Use the data to determine whether the recognition rate is greater for AREA B with a significance level of 5%.

<table>
<thead>
<tr>
<th></th>
<th>KNOW</th>
<th>DON'T KNOW</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREA A</td>
<td>130</td>
<td>90</td>
<td>220</td>
</tr>
<tr>
<td>AREA B</td>
<td>160</td>
<td>80</td>
<td>240</td>
</tr>
</tbody>
</table>

**Test** $H_0 : p_1 = p_2$ $H_1 : p_1 < p_2$
$\chi^2 = 0$ ? -

220 RUN

**Test** $H_0 : p_1 = p_2$ $H_1 : p_1 < p_2$
$\chi^2 = 0$ ? -

(Enter number of data from AREA A)
<table>
<thead>
<tr>
<th>Test</th>
<th>H₀: p₁ = p₂</th>
<th>H₁: p₁ &lt; p₂</th>
<th>(Enter number of KNOWs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>130</td>
<td></td>
<td></td>
<td>(Enter number of dat from AREA B)</td>
</tr>
<tr>
<td>240</td>
<td></td>
<td></td>
<td>(Enter number of KNOWs)</td>
</tr>
<tr>
<td>160</td>
<td></td>
<td></td>
<td>(Enter significance level α [%])</td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td>(Enter significance level; 5% is already set, so simply press 34)</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td>(Display test result)</td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test</th>
<th>H₀: p₁ = p₂</th>
<th>H₁: p₁ &lt; p₂</th>
<th>(Enter number of KNOWs)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Enter number of dat from AREA B)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Enter number of KNOWs)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Enter significance level α [%])</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Enter significance level; 5% is already set, so simply press 34)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Display test result)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Return to initial display)</td>
</tr>
</tbody>
</table>

Here, it is determined that the recognition rate in AREA B is greater than that in AREA A.
### 1 CHARACTER CODE TABLE

<table>
<thead>
<tr>
<th>HEX</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16</td>
<td>32</td>
<td>48</td>
<td>64</td>
<td>80</td>
<td>96</td>
<td>112</td>
<td>128</td>
<td>144</td>
<td>160</td>
<td>176</td>
<td>192</td>
<td>208</td>
<td>224</td>
<td>240</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>33</td>
<td>45</td>
<td>65</td>
<td>81</td>
<td>97</td>
<td>113</td>
<td>129</td>
<td>145</td>
<td>161</td>
<td>177</td>
<td>193</td>
<td>209</td>
<td>225</td>
<td>241</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>34</td>
<td>50</td>
<td>66</td>
<td>82</td>
<td>98</td>
<td>114</td>
<td>130</td>
<td>146</td>
<td>162</td>
<td>178</td>
<td>194</td>
<td>210</td>
<td>226</td>
<td>242</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>35</td>
<td>51</td>
<td>67</td>
<td>83</td>
<td>99</td>
<td>115</td>
<td>131</td>
<td>147</td>
<td>163</td>
<td>179</td>
<td>195</td>
<td>211</td>
<td>227</td>
<td>243</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>36</td>
<td>52</td>
<td>68</td>
<td>84</td>
<td>100</td>
<td>116</td>
<td>132</td>
<td>148</td>
<td>164</td>
<td>180</td>
<td>196</td>
<td>212</td>
<td>228</td>
<td>244</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>37</td>
<td>53</td>
<td>69</td>
<td>85</td>
<td>101</td>
<td>117</td>
<td>133</td>
<td>149</td>
<td>165</td>
<td>181</td>
<td>197</td>
<td>213</td>
<td>229</td>
<td>245</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>38</td>
<td>54</td>
<td>70</td>
<td>102</td>
<td>118</td>
<td>134</td>
<td>150</td>
<td>166</td>
<td>182</td>
<td>198</td>
<td>214</td>
<td>230</td>
<td>246</td>
<td>262</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>23</td>
<td>39</td>
<td>55</td>
<td>71</td>
<td>103</td>
<td>119</td>
<td>135</td>
<td>151</td>
<td>167</td>
<td>183</td>
<td>199</td>
<td>215</td>
<td>231</td>
<td>247</td>
<td>263</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>24</td>
<td>40</td>
<td>56</td>
<td>72</td>
<td>104</td>
<td>120</td>
<td>136</td>
<td>152</td>
<td>168</td>
<td>184</td>
<td>200</td>
<td>216</td>
<td>232</td>
<td>248</td>
<td>264</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>25</td>
<td>41</td>
<td>57</td>
<td>73</td>
<td>105</td>
<td>121</td>
<td>137</td>
<td>153</td>
<td>169</td>
<td>185</td>
<td>201</td>
<td>217</td>
<td>233</td>
<td>249</td>
<td>265</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>26</td>
<td>42</td>
<td>58</td>
<td>74</td>
<td>106</td>
<td>122</td>
<td>138</td>
<td>154</td>
<td>170</td>
<td>186</td>
<td>202</td>
<td>218</td>
<td>234</td>
<td>250</td>
<td>266</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>27</td>
<td>43</td>
<td>59</td>
<td>75</td>
<td>107</td>
<td>123</td>
<td>139</td>
<td>155</td>
<td>171</td>
<td>187</td>
<td>203</td>
<td>219</td>
<td>235</td>
<td>251</td>
<td>267</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>28</td>
<td>44</td>
<td>60</td>
<td>76</td>
<td>108</td>
<td>124</td>
<td>140</td>
<td>156</td>
<td>172</td>
<td>188</td>
<td>204</td>
<td>220</td>
<td>236</td>
<td>252</td>
<td>268</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>29</td>
<td>45</td>
<td>61</td>
<td>77</td>
<td>109</td>
<td>125</td>
<td>141</td>
<td>157</td>
<td>173</td>
<td>189</td>
<td>205</td>
<td>221</td>
<td>237</td>
<td>253</td>
<td>269</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>30</td>
<td>46</td>
<td>62</td>
<td>78</td>
<td>110</td>
<td>126</td>
<td>142</td>
<td>158</td>
<td>174</td>
<td>190</td>
<td>206</td>
<td>222</td>
<td>238</td>
<td>254</td>
<td>270</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>31</td>
<td>47</td>
<td>63</td>
<td>79</td>
<td>111</td>
<td>127</td>
<td>143</td>
<td>159</td>
<td>175</td>
<td>191</td>
<td>207</td>
<td>223</td>
<td>239</td>
<td>255</td>
<td>271</td>
<td></td>
</tr>
</tbody>
</table>

*Things to note:*
- The output for character codes is not specified; indicated by a blank cell in the table.
- Control codes are indicated by parentheses and are not displayed.
- Characters which cannot be input directly can be displayed using the CHRS function.
- Values in the lower right corner of each cell indicate the decimal value of the corresponding character code.
NOTE:
The special characters in the character code table below only appear on the display and are not printed out by the printer. When a LLIST or LPRINT command is executed, they are substituted by the differently shaped printer characters corresponding to the respective character codes. Refer to the pocket computer and printer character code tables and compare them for further details.

Character Code Table

| High-order digit | 0 | 16 | 32 | 48 | 64 | 80 | 96 | 112 | 128 | 144 | 160 | 176 | 192 | 208 | 224 | 240 |
|------------------|---|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Low-order digit  |   |    |    |    |    |    |    |     |     |     |     |     |     |     |     |     |     |
| 0                | 0 | 1  | 2  | 3  | 4  | 5  | 6  | 7   | 8   | 9   | A   | B   | C   | D   | E   | F   |
| 1                | 1 | (DEL) | 1 | A | Q | a | 0 | / | 1 | ア | チ | ム | 円 |    |    |    |    |
| 2                | 2 | (DEL) | 2 | B | R | b | r | 2 | ツ | メ | 年 |    |    |    |    |    |    |
| 3                | 3 |    | 3 | C | S | c | s | 3 | ワ | チ | モ | 甲 |    |    |    |    |    |
| 4                | 4 |    | 4 | D | T | d | t | 4 | プ | チ | リ | ￥ |    |    |    |    |    |
| 5                | 5 |    | 5 | E | U | e | u | 5 | オ | ナ | コ | ￥ |    |    |    |    |    |
| 6                | 6 |    | 6 | F | V | f | v | 6 | カ | ニ | ヨ | 万 |    |    |    |    |    |
| 7                | 7 |    | 7 | G | W | g | w | 7 | サ | ヌ | ラ | ￥ |    |    |    |    |    |
| 8                | 8 |    | 8 | H | X | h | x | 8 | エ | ナ | リ | ￥ |    |    |    |    |    |
| 9                | 9 |    | 9 | I | Y | i | y | 9 | コ | ノ | ゴ | ￥ |    |    |    |    |    |
| A                | A |    | A | J | Z | z |    | A | ニ | コ | ハ | レ |    |    |    |    |    |
| B                | B |    | B | K | l | l |    | B | サ | ヒ | ￥ |    |    |    |    |    |    |
| C                | C | ( ) | C | L | k | l |    | C | シ | フ | ￥ |    |    |    |    |    |    |
| D                | D | ( ) | D | M | m | ￥ |    | D | コ | ス | ケ | ￥ |    |    |    |    |    |
| E                | E | ( ) | E | N | n | ￥ |    | E | メ | テ | セ | ￥ |    |    |    |    |    |
| F                | F | ( ) | F | O | o | ￥ |    | F | マ | ￥ |    |    |    |    |    |    |    |

* Characters marked with a circle (○) are available on the PB-100 series.
* Except for the special characters, all can be printed with the FP-40 and FP-100.
* To display characters that cannot be input directly, use the respective CHR$ function.

Example: Display “Σ”

CAL mode: CHR$(132) □□ or CHR$(&H84) □□
BASIC mode: PRINT CHR$(132) □□ or PRINT CHR$(&H84) □□

* The hatched parts in the table (hexadecimal FC – FF (decimal 252 – 255)) are used internally and therefore not defined.
# 12-2 ERROR MESSAGE TABLE

<table>
<thead>
<tr>
<th>Error code</th>
<th>Error message</th>
<th>Meaning</th>
<th>Correction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OM error</td>
<td>a) Insufficient memory or system overflow.</td>
<td>a) Shorten program and check array dimensioning.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Erroneous CLEAR statement specification.</td>
<td>b) Check CLEAR statement value.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>c) Use RAM expansion pack.</td>
</tr>
<tr>
<td>2</td>
<td>SN error</td>
<td>Erroneous command or statement format.</td>
<td>a) Check spelling of commands.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>b) Check program input.</td>
</tr>
<tr>
<td>3</td>
<td>ST error</td>
<td>String length exceeds 255 characters.</td>
<td>Shorten string to 255 characters or less.</td>
</tr>
<tr>
<td>4</td>
<td>TC error</td>
<td>Formula too complex.</td>
<td>Divide formula into smaller sub-formulas</td>
</tr>
<tr>
<td>5</td>
<td>BV error</td>
<td>a) I/O buffer overflow.</td>
<td>a) Set RS-232C baud rate to lower value or set XON/OFF.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Line length exceed 255 bytes or 255 characters.</td>
<td>b) Keep lines 255 characters or less in length.</td>
</tr>
<tr>
<td>6</td>
<td>NR error</td>
<td>I/O device not ready for input/output.</td>
<td>Check connection and power switch of I/O device.</td>
</tr>
<tr>
<td>7</td>
<td>RW error</td>
<td>Error generated in I/O device operation.</td>
<td>Check I/O device.</td>
</tr>
<tr>
<td>8</td>
<td>BF error</td>
<td>Improper filename specification.</td>
<td>Check filename.</td>
</tr>
<tr>
<td>9</td>
<td>BN error</td>
<td>Improper file number specification.</td>
<td>Check file number.</td>
</tr>
<tr>
<td>13</td>
<td>OV error</td>
<td>Value exceeds allowable calculation result or input range.</td>
<td>Check values.</td>
</tr>
<tr>
<td>14</td>
<td>MA error</td>
<td>a) Mathematical error such as division by zero.</td>
<td>Check expressions and values.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Argument exceeds allowable calculation range.</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>DD error</td>
<td>Double declaration of identical array.</td>
<td>Either erase previous array or use a different array name.</td>
</tr>
<tr>
<td>16</td>
<td>BS error</td>
<td>Subscript or parameter outside of allowable range.</td>
<td>a) Check subscripts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>b) Increase size of arrays.</td>
</tr>
<tr>
<td>17</td>
<td>FC error</td>
<td>a) Erroneous use of function or statement.</td>
<td>a) Check argument values and statements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Illegal command used in direct mode or program mode.</td>
<td>b) Check for statements that can not be used in respective mode.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c) Illegal command used in CAL mode.</td>
<td>c) Check statements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d) Attempt to use undeclared array.</td>
<td>d) Declare array using DIM statement.</td>
</tr>
<tr>
<td>Error code</td>
<td>Error message</td>
<td>Meaning</td>
<td>Correction</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>18</td>
<td>UL error</td>
<td>a) Branch destination line number does not exist.</td>
<td>a) Check line numbers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Input of statement without line number in BASIC editing mode.</td>
<td>b) Always use line numbers in BASIC editing mode.</td>
</tr>
<tr>
<td>19</td>
<td>TM error</td>
<td>a) Mismatch of variable type and contents.</td>
<td>Check for illegal numeric assignment to string variables or string assignment to numeric variable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Mismatch of READ statement variable and data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c) Mismatch of INPUT# statement variable and data.</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>RE error</td>
<td>RESUME statement outside of error handling routine.</td>
<td>Check RESUME statement location.</td>
</tr>
<tr>
<td>21</td>
<td>PR error</td>
<td>Execution of command that cannot be used with password protected files.</td>
<td>Cancel password.</td>
</tr>
<tr>
<td>22</td>
<td>DA error</td>
<td>READ statement execution when no data present.</td>
<td>Check READ and DATA statements.</td>
</tr>
<tr>
<td>23</td>
<td>FO error</td>
<td>No FOR for NEXT statement.</td>
<td>Check for matching of FOR and NEXT statements.</td>
</tr>
<tr>
<td>24</td>
<td>NX error</td>
<td>No NEXT for FOR statement.</td>
<td>Check for matching of FOR and NEXT statements.</td>
</tr>
<tr>
<td>25</td>
<td>GS error</td>
<td>Mismatch of GOSUB and RETURN statements.</td>
<td>Check for matching of GOSUB and RETURN statements.</td>
</tr>
<tr>
<td>26</td>
<td>OP error</td>
<td>a) Attempt to access unopened file.</td>
<td>a) Execute OPEN statement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Attempt to open already opened file.</td>
<td>b) CLOSE file and then reopen.</td>
</tr>
<tr>
<td>29</td>
<td>AM error</td>
<td>Attempt to use output-related command for device opened for input or vice versa.</td>
<td>Ensure proper use of input-related and output-related commands.</td>
</tr>
<tr>
<td>30</td>
<td>FR error</td>
<td>Framing error detected by RS-232C port.</td>
<td>Check RS-232C connection and data transmission method.</td>
</tr>
<tr>
<td>31</td>
<td>PO error</td>
<td>Parity error or over run error detected by RS-232C port. Erro...</td>
<td>a) Check RS-232C connection and data transmission method.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>b) Adjust the playback volume of the tape recorder.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>c) Attempt using the phase which is opposite the current setting.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>d) Clean the head of the tape recorder.</td>
</tr>
</tbody>
</table>
### 12-3 COMMAND/FUNCTION TABLE

#### COMMANDS

<table>
<thead>
<tr>
<th>PASS</th>
<th>ON GOTO</th>
<th>BEEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEW (ALL)</td>
<td>ON GOSUB</td>
<td>INPUT</td>
</tr>
<tr>
<td>CLEAR</td>
<td>IF THEN ELSE</td>
<td>INKEY$</td>
</tr>
<tr>
<td>FRE</td>
<td>IF GOTO ELSE</td>
<td>INPUT$</td>
</tr>
<tr>
<td>LIST (ALL)</td>
<td>FOR NEXT</td>
<td>DIM</td>
</tr>
<tr>
<td>EDIT</td>
<td>REM</td>
<td>ERASE</td>
</tr>
<tr>
<td>VARLIST</td>
<td>LET</td>
<td>PEEK</td>
</tr>
<tr>
<td>RUN</td>
<td>DATA</td>
<td>POKE</td>
</tr>
<tr>
<td>TRON</td>
<td>READ</td>
<td>DEFSEG</td>
</tr>
<tr>
<td>TROFF</td>
<td>RESTORE</td>
<td>ON ERROR GOTO</td>
</tr>
<tr>
<td>END</td>
<td>PRINT</td>
<td>RESUME</td>
</tr>
<tr>
<td>STOP</td>
<td>TAB</td>
<td>ERL</td>
</tr>
<tr>
<td>GOTO</td>
<td>LOCATE</td>
<td>ERR</td>
</tr>
<tr>
<td>GOSUB</td>
<td>CLS</td>
<td></td>
</tr>
<tr>
<td>RETURN</td>
<td>SET</td>
<td></td>
</tr>
</tbody>
</table>

#### INPUT/OUTPUT COMMANDS

<table>
<thead>
<tr>
<th>LLIST</th>
<th>INPUT$</th>
<th>NEW #</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPRINT</td>
<td>EOF</td>
<td>LIST #</td>
</tr>
<tr>
<td>OPEN</td>
<td>SAVE (ALL)</td>
<td>LLIST #</td>
</tr>
<tr>
<td>CLOSE</td>
<td>LOAD (ALL)</td>
<td>SAVE #</td>
</tr>
<tr>
<td>PRINT #</td>
<td>VERIFY</td>
<td>LOAD #</td>
</tr>
<tr>
<td>INPUT #</td>
<td></td>
<td>READ #</td>
</tr>
</tbody>
</table>

#### DATA BANK COMMAND

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### FUNCTIONS

<table>
<thead>
<tr>
<th>ANGLE</th>
<th>HYPACS</th>
<th>FRAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIN</td>
<td>HYPATN</td>
<td>ROUND</td>
</tr>
<tr>
<td>COS</td>
<td>EXP</td>
<td>RAN #</td>
</tr>
<tr>
<td>TAN</td>
<td>LOG</td>
<td>PI</td>
</tr>
<tr>
<td>ASN</td>
<td>SQR</td>
<td>FACT</td>
</tr>
<tr>
<td>ACS</td>
<td>CUR</td>
<td>NCR</td>
</tr>
<tr>
<td>ATN</td>
<td>ABS</td>
<td>NPR</td>
</tr>
<tr>
<td>HYPSIN</td>
<td>SGN</td>
<td>POL</td>
</tr>
<tr>
<td>HYPCOS</td>
<td>INT</td>
<td>REC</td>
</tr>
<tr>
<td>HYPTAN</td>
<td>FIX</td>
<td>CHR$</td>
</tr>
<tr>
<td>HYPASIN</td>
<td></td>
<td>ASC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STRS</th>
<th>VAL</th>
<th>VALF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MID$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RIGHTS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LEFTS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&amp;H</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HEX$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DMS$</td>
</tr>
<tr>
<td>2ABS</td>
<td>2FACT</td>
<td>2ON</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>ACS</td>
<td>FIX</td>
<td>OPEN</td>
</tr>
<tr>
<td>ALL</td>
<td>FOR</td>
<td>OR</td>
</tr>
<tr>
<td>AND</td>
<td>FRAC</td>
<td>OUT</td>
</tr>
<tr>
<td>ANGLE</td>
<td>FRE</td>
<td></td>
</tr>
<tr>
<td>AS</td>
<td>GOSUB</td>
<td></td>
</tr>
<tr>
<td>ASC</td>
<td>GOTO</td>
<td></td>
</tr>
<tr>
<td>ASN</td>
<td>HEX$</td>
<td></td>
</tr>
<tr>
<td>ATN</td>
<td>HYP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BEEP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALC</td>
<td>IF</td>
</tr>
<tr>
<td></td>
<td>Chrs</td>
<td>INKEYS</td>
</tr>
<tr>
<td></td>
<td>CLEAR</td>
<td>INPUT</td>
</tr>
<tr>
<td></td>
<td>CLOSE</td>
<td>INT</td>
</tr>
<tr>
<td></td>
<td>CLS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COS</td>
<td>KEY</td>
</tr>
<tr>
<td></td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CUR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DATA</td>
<td>LEFTS</td>
</tr>
<tr>
<td></td>
<td>DEF</td>
<td>LEN</td>
</tr>
<tr>
<td></td>
<td>DEFM</td>
<td>LET</td>
</tr>
<tr>
<td></td>
<td>DEFSEG</td>
<td>LIST</td>
</tr>
<tr>
<td></td>
<td>DEG</td>
<td>LIST</td>
</tr>
<tr>
<td></td>
<td>DIM</td>
<td>LOAD</td>
</tr>
<tr>
<td></td>
<td>DM$</td>
<td>LOCATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LPRINT</td>
</tr>
<tr>
<td>2DATA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DEF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DEFM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DEFSEG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DEG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DIM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2DM$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2EDIT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ELSE</td>
<td></td>
<td>MID</td>
</tr>
<tr>
<td>END</td>
<td></td>
<td>MIDS</td>
</tr>
<tr>
<td>EOF</td>
<td></td>
<td>MOD</td>
</tr>
<tr>
<td>ERASE</td>
<td></td>
<td>MODE</td>
</tr>
<tr>
<td>ERL</td>
<td></td>
<td>MCR</td>
</tr>
<tr>
<td>ERR</td>
<td></td>
<td>NCR</td>
</tr>
<tr>
<td>ERROR</td>
<td></td>
<td>NEW</td>
</tr>
<tr>
<td>EXP</td>
<td></td>
<td>NEXT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NOT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NPR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TAN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>THEN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TO</td>
</tr>
</tbody>
</table>
SPECIFICATIONS

Model:
FX-850P

Basic calculation functions:
Negative numbers, exponents, parenthetical arithmetic operations (with priority sequence
judgment function—true algebraic logic), integer division, integer division remainders,
logical operators.

Built-in functions:
Trigonometric/inverse trigonometric functions (angle units: degrees, radians, grads),
logarithmic/exponential functions, square roots, cube roots, powers, hyperbolic/inverse
hyperbolic functions, conversion to integer, deletion of integer portion, absolute values,
signs, coordinate conversions, factorials, permutations, combinations, rounding, random
numbers, pi, decimal-sexagesimal conversions, decimal-hexadecimal conversions.

Number of built-in scientific library:
116

Commands:
EDIT, LIST, LLIST, LOAD, NEW, NEW ALL, RUN, SAVE, VERIFY, ANGLE, BEEP, CLEAR,
CLOSE, CLS, DEFSEG, DIM, ERASE, LET, LOCATE, LPRINT, PASS, POKE, PRINT, SET,
TROFF, TRON, VARLIST, DATA, END, FOR ~ NEXT ~ STEP, GOSUB ~ RETURN, GOTO,
IF ~ THEN ~ ELSE, INPUT, INPUT #, ON ~ ERROR ~ GOTO, ON ~ GOSUB,
ON ~ GOTO, OPEN, PRINT #, READ, REM, RESTORE, RESUME, RETURN, STOP,
LIST #, LLIST #, LOAD #, SAVE #, NEW #, READ #, RESTORE #, WRITE #

Program functions:
ASC( ), CHR$( ), INKEY$, INPUT$, LEFT$, LEN( ), MID$( ), RIGHT$( ), STR$( ),
TAB( ), VAL( ), VALF( )

Other functions:
EOF( ), ERL, ERR, PEEK( )

Calculation range:
$\pm 1 \times 10^{-99} \pm 9.999999999 \times 10^{99}$ and 0. Internal operation uses 12-digit mantissa.

Program system:
Stored system

Program language:
BASIC

RAM capacity:
Standard 8KB, expandable up to 40KB (Including 3KB in system area).

Number of program areas:
Maximum 10 (P0 through P9)

Number of stacks:
Subroutine: 96 levels
FOR ~ NEXT loop: 29 levels

Display contents:
10-digit mantissa + 2-digit exponent

Display elements:
32-column x 2-line dot matrix liquid crystal display
Main components:
   C-MOS VLSI and others

Power supply:
   2 lithium batteries (CR2032) for the mainframe
   1 lithium battery (CR1220) for memory backup

Power consumption:
   0.04W

Battery life:
   1. Continuous program execution: Approx. 90 hours
   2. Continuous display of 5555555555 at 20° C (68° F): Approx. 150 hours
   4.5 months when unit is used 1 hour per day.
* Note: 1 hour includes 10 minutes of condition 1 and 50 minutes of condition 2.

Memory protection battery:
   Approx. 2 years (with main batteries installed)

Auto power-off:
   Approximately 6 minutes

Ambient temperature range:
   0° C to 40° C (32° F to 104° F)

Dimensions:
   11.6 (H) × 193 (W) × 76 (D) mm
   (1/2" (H) × 7 5/8" (W) × 3" (D))

Weight:
   197g (6.9oz) including batteries.

Accessory:
   Hard case
# BASIC COMMAND INDEX

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td></td>
</tr>
<tr>
<td>&amp;H</td>
<td>155</td>
</tr>
<tr>
<td>ABS</td>
<td>137</td>
</tr>
<tr>
<td>ANGLE</td>
<td>129</td>
</tr>
<tr>
<td>ASC</td>
<td>147</td>
</tr>
<tr>
<td>ASN/ACS/ATN</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>B</strong></td>
<td></td>
</tr>
<tr>
<td>BEEP</td>
<td>117</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C</strong></td>
<td></td>
</tr>
<tr>
<td>CHRS</td>
<td>146</td>
</tr>
<tr>
<td>CLEAR</td>
<td>89</td>
</tr>
<tr>
<td>CLOSE</td>
<td>162</td>
</tr>
<tr>
<td>CLS</td>
<td>115</td>
</tr>
<tr>
<td>CUR</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D</strong></td>
<td></td>
</tr>
<tr>
<td>DATA</td>
<td>108</td>
</tr>
<tr>
<td>DEFSEG</td>
<td>125</td>
</tr>
<tr>
<td>DEG</td>
<td>156</td>
</tr>
<tr>
<td>DIM</td>
<td>121</td>
</tr>
<tr>
<td>DMSS</td>
<td>157</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>E</strong></td>
<td></td>
</tr>
<tr>
<td>EDIT</td>
<td>93</td>
</tr>
<tr>
<td>END</td>
<td>96</td>
</tr>
<tr>
<td>EOF</td>
<td>166</td>
</tr>
<tr>
<td>ERASE</td>
<td>122</td>
</tr>
<tr>
<td>ERL</td>
<td>128</td>
</tr>
<tr>
<td>ERR</td>
<td>128</td>
</tr>
<tr>
<td>EXP</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>F</strong></td>
<td></td>
</tr>
<tr>
<td>FACT</td>
<td>142</td>
</tr>
<tr>
<td>FIX</td>
<td>139</td>
</tr>
<tr>
<td>FOR - NEXT</td>
<td>104</td>
</tr>
<tr>
<td>FRAC</td>
<td>140</td>
</tr>
<tr>
<td>FRE</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>G</strong></td>
<td></td>
</tr>
<tr>
<td>GOSUB</td>
<td>99</td>
</tr>
<tr>
<td>GOTO</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>H</strong></td>
<td></td>
</tr>
<tr>
<td>HEX$</td>
<td>154</td>
</tr>
<tr>
<td>HYPASN/HYPACS/HYPATN</td>
<td>133</td>
</tr>
<tr>
<td>HYP SIN/HYP COS/HYP TAN</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>I</strong></td>
<td></td>
</tr>
<tr>
<td>IF THEN ELSE/</td>
<td></td>
</tr>
<tr>
<td>IF GOTO ELSE</td>
<td>103</td>
</tr>
<tr>
<td>INKEYS</td>
<td>119</td>
</tr>
<tr>
<td>INPUT</td>
<td>118</td>
</tr>
<tr>
<td>INPUT#</td>
<td>164</td>
</tr>
<tr>
<td>INPUT$</td>
<td>120, 165</td>
</tr>
<tr>
<td>INT</td>
<td>139</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>L</strong></td>
<td></td>
</tr>
<tr>
<td>LEFT$</td>
<td>153</td>
</tr>
<tr>
<td>LEN</td>
<td>154</td>
</tr>
<tr>
<td>LET</td>
<td>107</td>
</tr>
<tr>
<td>LIST [ALL]</td>
<td>92</td>
</tr>
<tr>
<td>LIST#</td>
<td>170</td>
</tr>
<tr>
<td>LLIST</td>
<td>158</td>
</tr>
<tr>
<td>LLIST#</td>
<td>171</td>
</tr>
<tr>
<td>LOAD, LOAD ALL</td>
<td>168</td>
</tr>
<tr>
<td>LOAD#</td>
<td>172</td>
</tr>
<tr>
<td>LOCATE</td>
<td>114</td>
</tr>
<tr>
<td>LOG/LN</td>
<td>135</td>
</tr>
<tr>
<td>LPRINT</td>
<td>159</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>M</strong></td>
<td></td>
</tr>
<tr>
<td>MIDS</td>
<td>151</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
</tr>
<tr>
<td>NCR</td>
<td>143</td>
</tr>
<tr>
<td>NEW [ALL]</td>
<td>89</td>
</tr>
<tr>
<td>NEW#</td>
<td>170</td>
</tr>
<tr>
<td>NPR</td>
<td>143</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>O</strong></td>
<td></td>
</tr>
<tr>
<td>ON ERROR GOTO</td>
<td>126</td>
</tr>
<tr>
<td>ON GOSUB</td>
<td>102</td>
</tr>
<tr>
<td>ON GOTO</td>
<td>101</td>
</tr>
<tr>
<td>OPEN</td>
<td>161</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>P</strong></td>
<td></td>
</tr>
<tr>
<td>PASS</td>
<td>88</td>
</tr>
<tr>
<td>PEEK</td>
<td>123</td>
</tr>
<tr>
<td>PI</td>
<td>142</td>
</tr>
<tr>
<td>POKE</td>
<td>124</td>
</tr>
<tr>
<td>POL</td>
<td>144</td>
</tr>
<tr>
<td>PRINT</td>
<td>111</td>
</tr>
<tr>
<td>PRINT#</td>
<td>163</td>
</tr>
<tr>
<td>R</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>---</td>
</tr>
<tr>
<td>RAN#</td>
<td>141</td>
</tr>
<tr>
<td>READ</td>
<td>109</td>
</tr>
<tr>
<td>READ#</td>
<td>173</td>
</tr>
<tr>
<td>REC</td>
<td>145</td>
</tr>
<tr>
<td>REM (')</td>
<td>106</td>
</tr>
<tr>
<td>RESTORE</td>
<td>110</td>
</tr>
<tr>
<td>RESTORE#</td>
<td>174</td>
</tr>
<tr>
<td>RESUME</td>
<td>127</td>
</tr>
<tr>
<td>RETURN</td>
<td>100</td>
</tr>
<tr>
<td>RIGHTS</td>
<td>152</td>
</tr>
<tr>
<td>ROUND</td>
<td>140</td>
</tr>
<tr>
<td>RUN</td>
<td>94</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SAVE, SAVE ALL</td>
<td>167</td>
</tr>
<tr>
<td>SAVE#</td>
<td>171</td>
</tr>
<tr>
<td>SET</td>
<td>116</td>
</tr>
<tr>
<td>SGN</td>
<td>138</td>
</tr>
<tr>
<td>SIN/COS/TAN</td>
<td>130</td>
</tr>
<tr>
<td>SQRT</td>
<td>136</td>
</tr>
<tr>
<td>STOP</td>
<td>97</td>
</tr>
<tr>
<td>STRS</td>
<td>148</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>T</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TAB</td>
<td>113</td>
</tr>
<tr>
<td>TROFF</td>
<td>95</td>
</tr>
<tr>
<td>TRON</td>
<td>95</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>V</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>VAL</td>
<td>149</td>
</tr>
<tr>
<td>VALF</td>
<td>150</td>
</tr>
<tr>
<td>VARLIST</td>
<td>93</td>
</tr>
<tr>
<td>VERIFY</td>
<td>169</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>W</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>WRITE#</td>
<td>175</td>
</tr>
</tbody>
</table>
GUIDELINES LAID DOWN BY FCC RULES FOR USE OF THE UNIT IN THE U.S.A.
(not applicable to other areas).
WARNING: This equipment generates and uses radio frequency energy and if not
installed and used properly, that is, in strict accordance with the manufacturer’s instruc-
tions, may cause interference to radio and television reception. It has been type tested
and found to comply with the limits for a Class B computing device in accordance with
the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide
reasonable protection against such interference in a residential installation. However,
there is no guarantee that interference will not occur in a particular installation. If this
equipment does cause interference to radio or television reception, which can be
determined by turning the equipment off and on, the user is encouraged to try to correct
the interference by one or more of the following measures:

. . . . reorient the receiving antenna
. . . . relocate the computer with respect to the receiver
. . . . move the computer away from the receiver
. . . . plug the computer into a different outlet so that computer and receiver are on
different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television tech-
nician for additional suggestions. The user may find the following booklet prepared by
the Federal Communications Commission helpful:
“How to Identify and Resolve Radio-TV Interference Problems”
This booklet is available from the US Government Printing Office, Washington D.C.,
20402, Stock No.004-000-00345-4.