GUIDELINES LAID DOWN BY FCC RULES FOR USE OF THE UNIT IN THE U.S.A. (not applicable to other areas).

NOTICE

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful 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 or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Caution: Changes or modifications to the product not expressly approved by CASIO could void the user's authority to operate the product.

CE

CASIO ELECTRONICS CO., LTD.
Unit 6, 1000 North Circular Road,
London NW2 7JD, U.K.

IMPORTANT!:
Please keep your manual and all information handy for future reference.

(NL)
Batterij niet weggooien, maar inleveren als KCA

DOT MATRIX LCD

fx-4800P

Contents

Handling Precautions

- Your calculator is made up of precision components. Never try to take it apart.
- Avoid dropping your calculator and subjecting it to strong impact.
- Do not store the calculator or leave it in areas exposed to high temperatures or humidity, or large amounts of dust. When exposed to low temperatures, the calculator may require more time to display results and may even fail to operate. Correct operation will resume once the calculator is brought back to normal temperature.
- The display will go blank and keys will not operate during calculations. When you are operating the keyboard, be sure to watch the display to make sure that all your key operations are being performed correctly.
- Replace batteries once every 5 years, regardless of how much the calculator is used during that period. Never leave dead batteries in the battery compartment. They can leak and damage the unit.
- Avoid using volatile liquids such as thinner or benzine to clean the unit. Wipe it with a soft, dry cloth, or with a cloth that has been dipped in a solution of water and a neutral detergent and wrung out.
- In no event will the manufacturer and its suppliers be liable to you or any other person for any damages, expenses, lost profits, lost savings or any other damages arising out of loss of data and/or formulas arising out of malfunction, repairs, or battery replacement. The user should prepare physical records of data to protect against such data loss.
- Never dispose of batteries, the liquid crystal panel, or other components by burning them.
- When the "Low battery!" message appears on the display, replace the main power supply batteries as soon as possible.
- Be sure that the power switch is set to OFF when replacing batteries.
- If the calculator is exposed to a strong electrostatic charge, its memory contents may be damaged or the keys may stop working. In such a case, perform the All Reset operation to clear the memory and restore normal key operation.
- Note that strong vibration or impact during program execution can cause execution to stop or can damage the calculator's memory contents.
- Using the calculator near a television or radio can cause interference with TV or radio reception.
- Before assuming malfunction of the unit, be sure to carefully reread this manual and ensure that the problem is not due to insufficient battery power, programming or operational errors.

Important

Before using the unit for the first time, be sure to load the batteries that come with it (page 8) and perform the RESET operation (page 11).

Be sure to keep physical records of all important data!
The large memory capacity of the unit makes it possible to store large amounts of data. You should note, however, that low battery power or incorrect replacement of the batteries that power the unit can cause the data stored in memory to be corrupted or even lost entirely. Stored data can also be affected by strong electrostatic charge or strong impact.

In no event shall CASIO Computer Co., Ltd. be liable to anyone for special, collateral, incidental, or consequential damages in connection with or arising out of the purchase or use of these materials. Moreover, CASIO Computer Co., Ltd. shall not be liable for any claim of any kind whatsoever against the use of these materials by any other party.

Power Supply

Power is supplied by two CR2032 lithium batteries. One battery (the main battery) powers normal operations, while the other battery (the back-up battery) provides the power required to retain data in memory.
The following message appears whenever main battery power gets too low:

Low battery!

Whenever this message appears, immediately turn off the calculator and replace the main battery as soon as possible.
If you continue to use the calculator after the low battery message appears, power will turn off automatically. Pressing the AC/ON button will not turn power back on if the main battery is too low. Also remember that even if you don't use the calculator, a low main battery can cause memory contents to be lost.

Important

- Normally, you should not remove both the main battery and the back-up battery from the calculator at the same time. Doing so can cause data stored in memory to become corrupted or lost entirely. If you do remove both batteries, correctly reload them and then perform the reset operation described on page 11 of this manual.
- Replace the batteries at least once every five years regardless of how much you use the calculator during that time.
- Remove both batteries from the calculator if you do not plan to use it for a long time.

A battery is loaded in the calculator at the factory for testing before shipment. Note that the service life provided by this test battery may be shorter than normal.

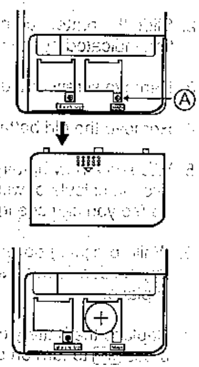
■ Replacing the Main Battery

Precautions

- Do not remove the back-up battery while the main battery is not loaded in the calculator.
- Be sure to turn the calculator off before replacing batteries. Leaving it on will cause data stored in memory to be lost.
- Never try to turn the calculator on while the main battery is not loaded or when it is loaded incorrectly. Doing so will cause data stored in memory to be lost and will cause malfunction of the calculator. Should this happen, remove the main battery and reload it correctly, and then perform the RESET operation described on page 11 of this manual.
- Replace the main battery at least once every five years to protect against damage caused by leaking battery fluid.

• To replace the main battery

1. Press SHIFT OFF to turn the calculator off.
2. Slide the battery compartment cover in the direction indicated by the arrow and remove it.
3. Remove screw Ⓐ and remove the battery holder.
4. Remove the old battery.
5. Wipe off a new battery with a dry cloth, and load it into the calculator with the positive "+" side facing up (so you can see it).
6. While pressing down on the battery with the battery holder, replace screw Ⓐ to secure the holder in place.
7. Replace the battery compartment cover, and then press AC/ON to turn on power.



- Memory contents are not lost when you replace the main battery as long as the back-up battery supplies power.
- If the figures on the display appear too light and hard to see after you turn on power, adjust the contrast using the procedure described on page 25 of this manual.

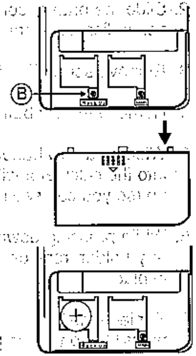
■ Replacing the Back-up Battery

Precautions

- Always check to make sure that the main battery is loaded and supplying power before removing the back-up battery from the calculator. If the low battery message (page 8) appears when you turn on the calculator, replace the main battery first and then replace the back-up battery.
- Do not remove the main battery while the back-up battery is not loaded in the calculator.
- Be sure to turn the calculator off before replacing batteries.
- Though the normal life of the back-up battery is five years, you should probably replace it more often in order to make sure you do not lose valuable data stored in memory.

• To replace the back-up battery

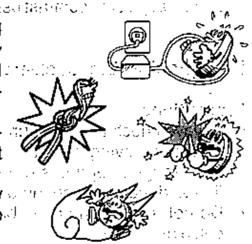
1. Press [SHIFT] [OFF] to turn the calculator off.
2. Slide the battery compartment cover in the direction indicated by the arrow and remove it.
3. Remove screw Ⓑ and remove the battery holder.
4. Remove the old battery.
5. Wipe off a new battery with a dry cloth, and load it into the calculator with the positive "+" side facing up (so you can see it).
6. While pressing down on the battery with the battery holder, replace screw Ⓑ to secure the holder in place.
7. Replace the battery compartment cover, and then press [AC/ON] to turn on power.



- Memory contents are not lost when you replace the back-up battery as long as the main battery supplies power.

Warning!
Improper use of batteries and leaking of battery fluid can stain and damage the calculator and other nearby items. It can also create the danger of fire or personal injury. Note the following important precautions concerning batteries.

- Always make sure that the positive "+" side of a battery is facing up (so you can see it) when you load it into the calculator.
- Never charge batteries, take them apart, or allow them to become shorted. Never expose batteries to direct heat or dispose of them by burning.



Keep batteries out of the reach of small children. If accidently swallowed, consult with a physician immediately.

■ About the Auto Power Off Function

The calculator switches power off automatically if you do not perform any key operation for about 6 minutes. To restore power, press [AC/ON].

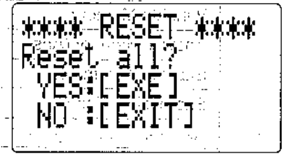
■ RESET Operation

The RESET operation returns the calculator to its original default settings. Remember that the RESET operation also deletes all data stored in calculator memory. If you need the data in memory, be sure to make a written record of it before performing the RESET operation.

• To reset the calculator

1. Press [MODE] [8] (RESET) and a RESET confirmation message appears on the display.

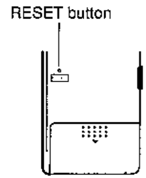
[MODE] [8] (RESET)



2. Press [EXE] to reset the calculator or [EXIT] to abort the RESET operation without changing anything.

[EXE]

- If the figures on the display appear too light or too dark after you perform the RESET operation, adjust the contrast using the procedure described on page 25 of this manual.
- If normal operation of the calculator is impossible for some reason, you can also start the RESET operation by using a thin, pointed object to press the RESET button on the back of the calculator. This causes the RESET confirmation screen to appear, so you can press [EXE] to reset or [EXIT] to abort.



Following are the original default settings that are obtained by resetting the calculator.

Item	Default Setting
Menu	COMP
Angle Measurement	Degrees (Deg)
Norm	Norm1
Number System	Decimal (Dec)
Variable Memory	Cleared
Answer Memory (Ans)	Clear
Statistical Memory	Clear
Expression Memory	Clear
Recursion Memory	Clear
Program Memory	Clear
Input Buffer/AC Replay	Clear

Important

If you perform the RESET operation while the calculator is performing an internal operation (indicated when power is on but there is nothing on the display), data being used by the operation will also be cleared. Always make sure that there is no internal operation being performed before performing a RESET operation.

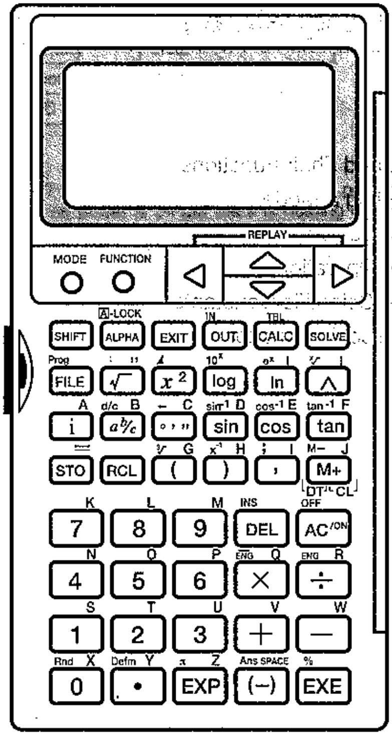
Chapter 1

Getting Acquainted

Chapter 1 Getting Acquainted

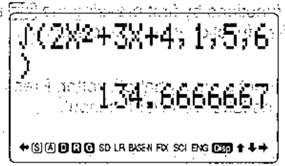
This chapter gives you a general introduction to the various capabilities of the unit. It contains important information about the unit, so you should be sure to read it before starting operation.

1-1 Keys and Their Functions



■ Display Indicators

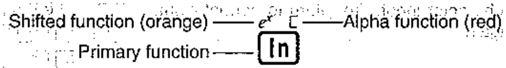
Indicators appear on the display to keep you informed about the current operational status of the calculator.



Indicator	Meaning
Ⓢ	Appears when SHIFT key is pressed to indicate keys will input functions marked in orange.
Ⓐ	Appears when ALPHA key is pressed to indicate keys will input letters marked in red.
D	Degrees selected as the unit of angular measurement.
R	Radians selected as the unit of angular measurement.
G	Grads selected as the unit of angular measurement.
SD	Calculator is in the SD Mode.
LR	Calculator is in the LR Mode.
BASE-N	Calculator is in the BASE-N Mode.
FIX	Number of decimal places specification is in effect.
SCI	Number of significant digits specification is in effect.
ENG	Engineering notation is in effect.
Disp	Displayed values is an intermediate result.
↑ ↓	Appears during display of a list to indicate data above or below the current screen.
← →	Indicates data runs off left or right of the current screen.

■ The Keyboard

Many of the unit's keys are used to perform more than one function. The functions marked on the keyboard are color coded to help you find the one you need quickly and easily.



• **Primary Functions**
These are the functions that are normally executed when you press the key.

• **Shifted Functions**
You can execute these functions by first pressing the [SHIFT] key, followed by the key that is assigned the shifted function you want to execute.

• **Alpha Functions**
An alpha function is the input of an alphabetic letter. Press the [ALPHA] key, followed by the key that is assigned the letter you want to input.

Alpha Lock

Normally, once you press [ALPHA] and then a key to input an alphabetic character, the keyboard reverts to its primary functions immediately. If you press [SHIFT] and then [ALPHA], the keyboard locks in alpha input until you press [ALPHA] again.

■ Key Operations

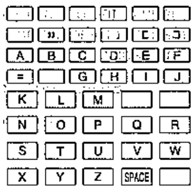
MODE
Mode Key
• Press this key to display the Main Menu. You can then input a number from 1 to 8 to select a mode. See "Selecting a Mode" on page 20 for details.

FUNCTION
Function Key
• Press this key to display the function menu.

REPLAY
[◁][⇳][▷] Cursor/Replay Keys
• Use these keys to move the cursor on the display.
• After you press the [EXE] key following input of a calculation or value, press [◀] to display the calculation from the end, or [▶] to display it from the beginning. You can then execute the calculation again, or edit the calculation and then execute it. See page 30 for details on the Replay Function.

[SHIFT] Shift Key
• Press this key to shift the keyboard and access the functions marked in orange. The [S] indicator on the display indicates that the keyboard is shifted. Pressing [SHIFT] again unshifts the keyboard and clears the [S] indicator from the display.

[A]-LOCK
[ALPHA] Alpha Key
• Press this key to input a letter marked in red on the keyboard.
• Press this key following [SHIFT] to lock the keyboard into alphabetic character input. To return to normal input, press [ALPHA] again.



[EXIT] Exit Key
• Press this key to quit a function menu, program input display, formula storage, table function, solve function, or recursion function.

IN
[OUT] In/Out Key
• Use this key when performing calculations using formula storage. See "Formula Storage" on page 108 for details.

TBL
[CALC] Formula Storage Function/Table Key
• Use this key when performing calculations using formula storage. See "Formula Storage" on page 108 for details.
• Press [SHIFT] [TBL] to define the range (variable conditions) for one of the variables in a formula storage expression. See "Formula Storage" on page 108 for details.

[SOLVE] Solve Key
• Use this key with formula storage to solve for a variable using Newton's method. See "Solve Function" on page 112 for details.

Prog
[FILE] File/Program Command Key
• Use this key to recall a specific file.
• In the COMP, BASE-N, SD, and LR Modes, input the following to run a program:

[SHIFT] [Prog] "file name" [EXE]

See "Running a Program" on page 121 for details.

[√] Square Root/Multistatement Key
• Press this key and then enter a value to obtain the square root of the value.
• Press this key following [SHIFT] to separate formulas or commands in programmed calculations or consecutive calculations. The result of such combinations is known as a "multistatement". For details, see page 28.

◢
[x²] Square/Display Key
• Enter a value and press this key to square the value.
• Press this key following [SHIFT] to display results of program calculations and consecutive calculations.

10ˣ
[log] Common Logarithm/Antilogarithm Key
• Press this key and then enter a value to obtain the common logarithm of the value.
• Press [SHIFT] [10ˣ] and then enter a value to make the value an exponent of 10.

[ln] Natural Logarithm/Exponential Key
• Press this key and then enter a value to obtain the natural logarithm of the value.
• Press [SHIFT] [eˣ] and then enter a value to make the value an exponent of e.
• Press this key following [ALPHA] to enter the open bracket [.

[∧] Power/Root Key
• Enter a value for x, press this key, and then enter a value for y to obtain x to the power of y.
• Enter a value for x, press [SHIFT] [ˣ√], and then enter a value for y to obtain the xth root of y.
• Press this key following [ALPHA] to enter the closed bracket].

[i] Imaginary Number Input Key
- Use this key to input imaginary number unit *i* for a complex number.
- Press this key in the BASE-N Mode to input the hexadecimal value A_{16}.

[a b/c] Fraction Key
- Use this key when entering fractions and mixed fractions. To enter the fraction 23/45, for example, press 23 [a b/c] 45. To enter 2-3/4, press 2 [a b/c] 3 [a b/c] 4.
- Press [SHIFT] [d/c] to display an improper fraction.
- Press this key in the BASE-N Mode to input the hexadecimal value B_{16}.

[°'"] Decimal ↔ Sexagesimal Key
- Press this key to enter a sexagesimal value: (degree/minute/second or hour/minute/second)

Example 78°45'12" ➔ 78 [°'"] 45 [°'"] 12 [°'"]

- When pressed following [SHIFT], a decimal based value can be displayed in degrees/minutes/seconds.
- Press this key in the BASE-N Mode to input the hexadecimal value C_{16}.

[sin] Sine Key
- Press this key and then enter a value to obtain the sine of the value.
- Press [SHIFT] [sin] and then enter a value to obtain the inverse sine of the value.
- Press this key in the BASE-N Mode to input the hexadecimal value D_{16}.

[cos] Cosine Key
- Press this key and then enter a value to obtain the cosine of the value.
- Press [SHIFT] [cos] and then enter a value to obtain the inverse cosine of the value.
- Press this key in the BASE-N Mode to input the hexadecimal value E_{16}.

[tan] Tangent Key
- Press this key and then enter a value to obtain the tangent of the value.
- Press [SHIFT] [tan] and then enter a value to obtain the inverse tangent of the value.
- Press this key in the BASE-N Mode to input the hexadecimal value F_{16}.

[STO] Store Key
- Press this key and then input a letter to store a calculation result to the variable specified by the letter.

[RCL] Recall Key
- Press this key and then input a letter to recall the value assigned to the variable specified by the letter.

[(] Open Parenthesis/Cube Root Key
- Press this key to enter an open parenthesis in a formula.
- Press [SHIFT] [∛] and then enter a value to obtain the cube root of the value.

[)] Close Parenthesis/Reciprocal Key
- Press this key to enter a close parenthesis in a formula.
- Enter a value and then press [SHIFT] [x^{-1}] to obtain the reciprocal of the value.

[,] Comma/Semicolon Key
- Press this key to input a comma.
- Press this key following [SHIFT] to enter a semicolon.

[M+] Memory Plus/Memory Minus/Data Input/Clear Key
- Press this key to add the displayed value to memory. Note that when formula is displayed, results are first derived and then stored into memory.
- Press this key following [SHIFT] to subtract the displayed value from memory.
- In the SD and LR modes, press this key to input data.
- In the SD and LR modes, press this key following [SHIFT] to clear data which has been input incorrectly.

[0] ~ [9], [•] 10-key Pad
- Use these keys to input values from left to right. Use [•] to input a decimal point. you can input up to 10 digits.
- Following operation of the [SHIFT] key, the menus marked in green (or orange) above these keys are accessed.

[SHIFT] [Rnd] Internal Rounding
This key operation rounds the internal value to 10 digits. Note that this also rounds the result that is produced by the Ans function. In the FIX and SCI modes, this key operation changes the internal value to the form specified for value display.

[SHIFT] [Defm] Memory Expansion
Use this key operation to expand the number of variables from the standard 26.

[DEL] Delete/Insert Key
- Press this key to delete the character at the current cursor location.
- Press [SHIFT] [INS] to display the insert cursor([ ]). You can insert characters or commands while the insert cursor is displayed.

[AC/ON] All Clear/ON/OFF Key
- Press this key to switch power on.
- Press this key while power is on to clear the display.
- Press this key following [SHIFT] to switch power off.

[+] [−] [×] [÷] Arithmetic Operator/Engineering Keys
- Use these keys to input arithmetic operators.
- Press [−] before inputting a value to indicate that the value is negative.
- The [×] and [÷] keys have the following operations with the [SHIFT] key.

[SHIFT] [ENG] Engineering Notation Conversion Key
This converts the displayed value to exponential notation in which the exponent is a positive value that is a multiple of three.

Examples 10^3 = k (kilo); 10^6 = M (mega); 10^9 = G (giga)

[SHIFT] [ENG] Engineering Notation Conversion Key
This converts the displayed value to exponential notation in which the exponent is a negative value that is a multiple of three.

Examples 10^{-3} = m (milli); 10^{-6} = μ (micro); 10^{-9} = n (nano); 10^{-12} = p (pico)

π Z
[EXP] Exponent/Pi Key
- Use this key when entering a mantissa and exponent. To input 2.56×10^{34}, for example, enter 2.56 [EXP] 34. Note that the maximum value that can be used for an exponent is ± 99. Any value outside this range results in a syntax error (Syn ERROR).
- Press [SHIFT] [π] to input the value of π.

Ans SPACE
[(–)] (–)/Answer/Space Key
- Press this key when entering a negative value.
- Press [SHIFT] and then this key to recall the most recent calculation result obtained using the [EXE] key.
- Press [ALPHA] and then this key to enter a space.

%
[EXE] Execute/Percent Key
- Press this key to obtain the result of a calculation. You can press this key following data input, or after a result is obtained to execute the calculation again using the previous result.
- Press this key following [SHIFT] for percentage calculations.

1-2 Selecting a Mode

Before you perform a calculation, you should first select the appropriate mode.

• To select a mode

1. Press the [MODE] key to display the Main Menu.

[MODE]

```
1.COMP  2.BASE-N
3.SD    4.LR
5.PROG  6.an
7.CONT  8.RESET
```

2. Input the number from 1 through 8 that corresponds to the mode you want to enter.

The following table describes the purpose of each mode.

Mode	Purpose
COMP	General calculations, including function calculations
BASE-N	Binary, octal, decimal, hexadecimal conversions and logical operations
SD	Single-variable (standard deviation) statistical calculations
LR	Paired-variable (regression) statistical calculations
PROG	File name assignment, program input, program execution in the program area
a_n	Recursion calculations
CONT	Display contrast adjustment
RESET	Reset operation

1-3 Basic Set Up

This section tells you how to perform the basic set up required by the calculator.

■ Function Menus

Before actually using this calculator to perform calculations, you should first specify the correct unit of angular measurement and the display format. To do this, press the [FUNCTION] key to display the function menu.

Example 1 **Function menu in the COMP Mode**

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG  6.DSP/CLR
```

Example 2 **Function menu in the SD/LR Mode**

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG  6.DSP/CLR
7.STAT 8.RESULTS
```

The items that make up a function menu depend on the mode the calculator is in when you press the [FUNCTION] key. Example function menu displays are shown the sections of this manual that describe each mode.
Following is a brief description of all the items that can appear in a function menu.

"1. MATH" .... Built-in Function Menu (page 30)
For recall of function commands not printed on keys or the key panel.

"2. COMPLX" ...... Complex Number Calculation Menu (page 74)
For recall of commands used in complex number calculations.

"3. PROG" .... Program Command Menu (page 132)
For insertion of special program command.

"4. CONST" .... Scientific Constant Menu (page 38)
For recall of scientific constants.

"5. DRG" .... Unit of Angular Measurement Menu (page 22)
For specification of the unit of angular measurement.

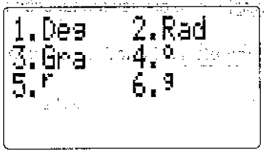
"6. DSP/CLR" ...... Display Format/Clear Menu (page 23)
For specification of the number of display digits and for switching engineering notation on and off. Also used to specify a memory area and clear its contents.

"7. STAT" .... Statistical Calculation Menu (page 91)
For recall of commands used in statistical processing of data.

"8. RESULTS" ...... Statistical Result Menu (page 91)
For display of calculation results produced by single-variable or paired-variable statistical calculations.

■ Unit of Angular Measurement (DRG) Menu

[5] (DRG)



"1. Deg" .... Specifies degrees as the default.
"2. Rad" .... Specifies radians as the default.
"3. Gra" .... Specifies grads as the default.
"4. o" .... Specifies degrees for a specific input value.
"5. r" .... Specifies radians for a specific input value.
"6. g" .... Specifies grads for a specific input value.

- The relationship between the angular measurement units is shown below.
  360° = 2π radians = 400 grads
  90° = π/2 radians = 100 grads

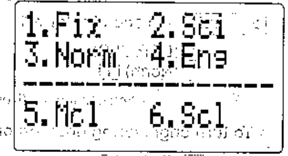
Example To specify degrees as the default unit of angular measurement.

[5] (DRG) [1] (Deg)

D
Display indicator for degrees

■ Display Format/Clear (DSP/CLR) Menu

[6] (DSP/CLR)



"1. Fix" .... Specifies number of decimal places for display.
"2. Sci" .... Specifies number of significant digits for display.
"3. Norm" .... Specifies range for switching to exponential format.
"4. Eng" .... Displays calculation results using engineering notation.
"5. Mcl" .... Clears all variables.
"6. Scl" .... Clears statistical memory.

• To specify the number of decimal places (Fix)

Example **To specify two places to the right of the decimal point.**

[6] (DSP/CLR)
[1] (Fix) [2]

You can input a number from 0 to 9.

FIX
Indicates a number of decimal places specification is in effect.

- Calculation results are rounded off to the number of decimal places you specify.
- The number of decimal places you specify remains in effect until you change the exponential display range (Norm) specification.

• To specify the number of significant digits (Sci)

Example **To specify three significant digits.**

[6] (DSP/CLR)
[2] (Sci) [3]

You can input a number from 0 to 9.

SCI
Indicates a number of significant digits specification is in effect.

- Calculation results are rounded off to the number of significant digits you specify.
- Inputting 0 specifies 10 as the number of significant digits.
- The number of significant digits you specify remains in effect until you change the exponential display range (Norm) specification.
- Even after you specify the number of decimal places or the number of significant digits, the calculator continues to use a 15-digit mantissa for internal calculations.

Whenever you want to round off the internal value to match your specifications, press [SHIFT] [Rnd].

- **To specify the exponential notation range (Norm 1/Norm 2)**

You can specify either Norm 1 or Norm 2 as the exponential notation range.

Norm 1 .... Exponential notation is automatically used for values less than 10^{-2} and values 10^{10} or greater.

Norm 2 .... Exponential notation is automatically used for values less than 10^{-9} and values 10^{10} or greater.

Example To specify Norm 1.

[6] (DSP/CLR)
[3] (Norm) [1]

You can input 1 (Norm 1) or 2 (Norm 2).

- **To turn engineering notation on and off (Eng)**

[6] (DSP/CLR)
[4] (Eng)

ENG

Indicates engineering notation is on.

- Each time you perform the above operation, the calculator switches between engineering notation and normal (non-engineering) notation.
- The following is a list of the engineering notation symbols and their values.

Symbol	Meaning	Unit
T	tera	10^{12}
G	giga	10^{9}
M	mega	10^{6}
k	kilo	10^{3}
m	milli	10^{-3}
μ	micro	10^{-6}
n	nano	10^{-9}
p	pico	10^{-12}
f	femto	10^{-15}

- The unit automatically selects the engineering symbol that makes the numeric value fall within the range of 1 to 999.

- **To clear all variables (A through Z)**

[6] (DSP/CLR)
[5] (Mcl) [EXE]

Mcl
0

- The above operation clears all standard variables (A through Z) and any other variables created by memory expansion.

- **To clear only statistical memories (P, Q, R, U, V, W)**

[6] (DSP/CLR)
[6] (Scl) [EXE]

Scl
0

- The above operation clears the U, V, and W variables used in the SD Mode.

■ Adjusting the Display Contrast

Use the following procedure to make the figures on the display lighter or darker.

1. While the Main Menu (page 20) is on the display, press [7] (CONT).

[MODE] [7] (CONT)

*** CONTRAST ***
LIGHT DARK
[←] [→]

2. Use [◄] and [►] to adjust the display contrast.
- [◄] makes figures lighter.
- [►] makes figures darker.
- You can hold down either arrow key for repeat operation.

3. After setting the contrast, press [MODE] to return to the Main Menu.

1-4 Basic Operation

The operations described here are fundamental calculations that you need to get started with the unit. Programming and statistical calculations are covered in their own separate sections.

■ Inputting Calculations

When you are ready to input a calculation, first press [AC] to clear the display. Next, input your calculation formulas exactly as they are written, from left to right, and press [EXE] to obtain a result.

Example 2(5 + 4) ÷ (4 × 3) =

[AC] [2] [(] [5] [+] [4] [)] [÷]
[(] [4] [×] [3] [)] [EXE]

2(5+4)÷(4×3)
1.5

The unit uses two types of functions: Type A functions and Type B functions. With Type A functions, you press the function key after you enter a value. With Type B functions, you press the function key first and then enter a value.

Type A function

	Example	Key Operation
Squares:	4^2	[4] [x^2]

Type B function

	Example	Key Operation
Sine:	2 sin45°	[2] [sin] [4] [5]

- For detailed examples on all of the possible calculations available, see the section titled "Calculation Priority Sequence" on page 41.

• To clear an entire calculation and start again

Press the [AC] key to clear the error along with the entire calculation. Next, re-input the calculation from the beginning.

■ Editing Calculations

Use the [◄] and [►] keys to move the cursor to the position you want to change, and then perform one of the operations described below. After you edit the calculation, you can execute it by pressing [EXE], or use [►] to move to the end of the calculation and input more.

• To change a step

Example To change cos60 to sin60

Key	Display
[cos] [6] [0]	cos 60_
[◄] [◄] [◄]	cos 60
[sin]	sin 60

• To delete a step

Example To change 36 × × 2 to 36 × 2

Key	Display
[3] [6] [×] [×] [2]	36××2_
[◄] [◄] [DEL]	36×2

• To insert a step

Example To change 2^2 to $\sin 2^2$

Key	Display
[2] [x^2]	2^2_
[◄] [◄]	2^2
[SHIFT] [INS]	2^2
[sin]	sin 2^2

- When you press [SHIFT] [INS] a space is indicated by the symbol "[]". The next function or value you input is inserted at the location of "[]". To abort the insert operation without inputting anything, move the cursor, press [SHIFT] [INS] again, or press [◄], [►], [EXE].

• To make corrections in the original calculation

Example 14 ÷ 0 × 2.3 entered by mistake for 14 ÷ 10 × 2.3

Key	Display
[AC] [1] [4] [÷] [0] [×] [2] [•] [3] [EXE]	14÷0×2.3 Ma ERROR
Press [◄] or [►].	14÷0×2.3
	Cursor is positioned automatically at the location of the cause of the error.
Make necessary changes. [◄] [SHIFT] [INS] [1]	14÷10×2.3
Execute it again. [EXE]	14÷10×2.3 32.2

■ Answer Function

The unit's Answer Function automatically stores the last result you calculated by pressing [EXE] (unless the [EXE] key operation results in an error). The result is stored in the answer memory.

• To recall the contents of the answer memory

[SHIFT] [Ans] [EXE]

• **To use the contents of the answer memory in a calculation**

Example 123 + 456 = 579
789 − 579 = 210

[AC] [1] [2] [3] [+] [4] [5] [6] [EXE]

```
123+456
                579
```

[7] [8] [9] [−] [SHIFT] [Ans] [EXE]

```
789-Ans
                210
```

- The largest value that the answer memory can hold is one with 15 digits for the mantissa and 2 digits for the exponent.
- Answer memory contents are not cleared when you press the [AC] key or when you switch power off.
- Operation of [EXE], [%], [M+], [SHIFT] [M−], or [STO] followed by a variable name (A to Z) automatically updates answer memory contents with the result of the operation.
- Contents of answer memory are not altered when [RCL] α (α = A to Z) is used to recall contents of variable memory. Also, contents of answer memory are not altered when variables are input when the variable input prompt is displayed.
- Whenever an operation produces an error, answer memory retains the last valid result produced.

■ Using Multistatements

Multistatements are formed by connecting a number of individual statements for sequential execution. You can use multistatements in manual calculations and in programmed calculations. There are two different ways that you can use to connect statements to form multistatements.

• **Colon (:)**
Statements that are connected with colons are executed from left to right, without stopping.

• **Display Result Command (◢)**
When execution reaches the end of a statement followed by a display result command, execution stops and the result up to that point appears on the display. You can resume execution by pressing the [EXE] key.

• **To use multistatements**

Example 6.9 × 123 = 848.7
123 ÷ 3.2 = 38.4375

[AC] [1] [2] [3] [STO] [A]
[6] [•] [9] [×] [ALPHA] [A] [SHIFT] [◢]
[ALPHA] [A] [÷] [3] [•] [2] [EXE]

```
                123
6.9×A◢
A÷3.2
              848.7
                Disp
```

Appears on display when "◢" is used.

[EXE]

```
                123
6.9×A◢
A÷3.2
            38.4375
```

- Note that the final result of a multistatement is always displayed, regardless of whether it ends with a display result command.
- You cannot construct a multistatement in which one statement directly uses the result of the previous statement.

Example 123 × 456: × 5
Invalid

■ Multiplication Operations without a Multiplication Sign

You can omit the multiplication sign (×) in any of the following operations.

- Before the type B functions (page 41)

Example **2sin30, 10log1.2, 2√3, 2pol(5, 12), etc.**

- Before constants, variable names, value memory names

Example **2π, 2AB, 3Ans, etc.**

- Before an open parenthesis

Example **3(5 + 6), (A + 1)(B − 1), etc.**

■ Performing Continuous Calculations

The unit lets you use the result of one calculation as one of the arguments in the next calculation. When performing such a calculation, the contents of the Ans Memory (which contains the result of the last calculation performed) are used in the next calculation.

Example 1 ÷ 3 =
1 ÷ 3 × 3 =

[AC] [1] [÷] [3] [EXE]

```
1÷3
       0.3333333333
```

(Continuing)
[×] [3] [EXE]

```
Ans×3
                  1
```

Continuous calculations can also be used with Type A functions (see page 41).

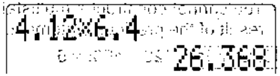
■ Using the Replay Function

The Replay Function automatically stores the last calculation performed into replay memory. You can recall the contents of the replay memory by pressing [◄] or [►]. If you press [►], the calculation appears with the cursor at the beginning. Pressing [◄] causes the calculation to appear with the cursor at the end. You can make changes in the calculation as you wish and then execute it again.

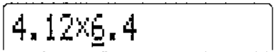
Example **To perform the following two calculations**

4.12 × 6.4 = 26.368
4.12 × 7.1 = 29.252

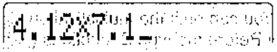
[AC] [4] [•] [1] [2] [×] [6] [•] [4] [EXE]



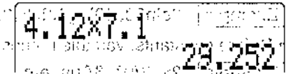
[◄] [◄] [◄] [◄]



[7] [•] [1]



[EXE]

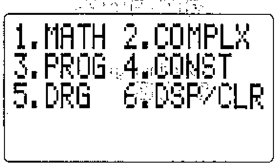


- The maximum capacity of the replay memory is 127 bytes. A calculation remains stored in replay memory until you perform another calculation or change modes.
- The contents of the replay memory are not cleared when you press the [AC] key, so you can recall a calculation and execute it even after performing the all clear operation. Note, however, that replay memory contents are cleared whenever you change to another mode or menu.

■ Built-in Function (MATH) Menu

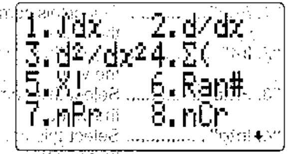
The MATH Menu can be used in the COMP, SD, LR, and a_n modes. It provides you with built-in scientific functions in addition to those available by pressing keys on the calculator's keyboard. Note that there is a total of four MATH Menu screens. Use the [▼] and [▲] keys to scroll between menus.

[FUNCTION]

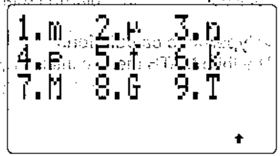


(In COMP Mode)

[1] (MATH)

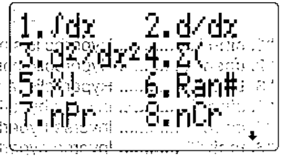


[▼] ↕ [▲]



• Integration, differential, Σ, probability

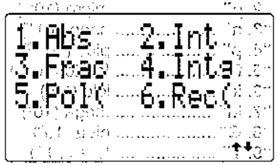
The first MATH menu provides tools for integrations, differentials and quadratic differentials, Σ (sigma) calculations, permutations, combinations, factorials, and random number generation.



"1. ∫*dx*" .... Integration (page 67)
"2. *d*/*dx*" .... Differential (page 62)
"3. d^2/dx^2" .... Quadratic differential (page 65)
"4. Σ(" .... Σ calculation (page 70)
"5. *x*!" .... Input a value and select this item to obtain the factorial of the value.
"6. Ran#" .... Generates a pseudo random number in the range of 0 to 1 (10 decimal places).
"7. *n*P*r*" .... Permutation
"8. *n*C*r*" .... Combination

• Numerical calculations

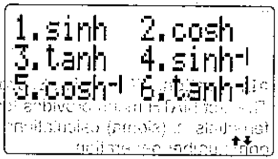
The second MATH menu includes items for absolute value calculations, integer and fraction extraction, and conversion between rectangular and polar coordinates.



"1. Abs" .... Select this item and input a value to obtain the absolute value of the value.
"2. Int" .... Select this item and input a value to extract the integer part of the value.
"3. Frac" .... Select this item and input a value to extract the fraction part of the value.
"4. Intg" .... Select this item and input a value to obtain the largest integer that is not greater than the value.
"5. Pol(" .... Rectangular-to-polar coordinate conversion
"6. Rec(" .... Polar-to-rectangular coordinate conversion

• Hyperbolic calculations

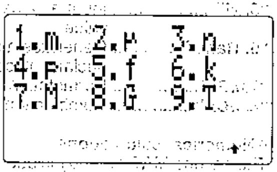
The third MATH menu contains hyperbolic and inverse hyperbolic functions.



"1. sinh" .... Hyperbolic sine of a value
"2. cosh" .... Hyperbolic cosine of a value
"3. tanh" .... Hyperbolic tangent of a value
"4. $sinh^{-1}$" .... Inverse hyperbolic sine of a value
"5. $cosh^{-1}$" .... Inverse hyperbolic cosine of a value
"6. $tanh^{-1}$" .... Inverse hyperbolic tangent of a value

• Engineering notation

The fourth MATH Menu provides a list of symbols for input of values using engineering notation.



"1. m" .... milli (10^{-3})
"2. μ" .... micro (10^{-6})
"3. n" .... nano (10^{-9})
"4. p" .... pico (10^{-12})
"5. f" .... femto (10^{-15})
"6. k" .... kilo (10^{3})
"7. M" .... mega (10^{6})
"8. G" .... giga (10^{9})
"9. T" .... tera (10^{12})

■ Memory

In the standard configuration, this calculator provides memory for 26 variables, which are named using alpha characters A through Z. Values assigned to memory can have up to 15 digits for the mantissa and up to two digits for the exponent. Values assigned to variables are retained even when you turn the calculator off.

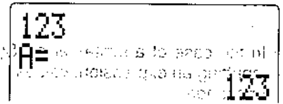
- Variable M is also used by the calculator as its the "independent memory," which is the one affected by [M+] and [SHIFT] [M-] operations.

• Using variables

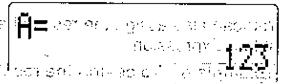
You can store values to up to 26 different variables for instant recall when you need them.

Example 1 **To assign the value 123 to variable A and then recall it.**

[AC] [1] [2] [3] [STO] [A]



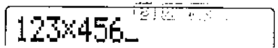
[AC] [RCL] [A]



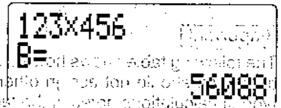
- If you input an expression in place of the value, the calculated result of the expression is assigned to the variable.

Example 2 **To assign the result of 123 × 456 to variable B.**

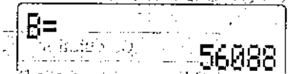
[AC] [1] [2] [3] [×] [4] [5] [6]



[STO] [B]



[AC] [RCL] [B]



- Once a variable is assigned a value, the variable name (alpha character) can be used in place of the value in an expression.

Example 3 **To multiply the value assigned to variable A in Example 1 by the value assigned to variable B in Example 2 and assign the result to variable C.**

[AC] [ALPHA] [A] [×] [ALPHA] [B] — A×B_

[STO] [C] — A×B / C= / 6898824

[AC] [RCL] [C] — C= / 6898824

- In the case of a syntax error (Syn ERROR) caused by making a mistake when inputting an expression, values assigned to the variables prior to the error remain unchanged.

You can also assign the result of a calculation to a variable using the format "Variable = Expression."

Example 4 **To assign the result of the expression log 2 to variable S.**

[AC] [ALPHA] [S] [ALPHA] [=] [log] [2] [EXE] — S=log 2 / 0.3010299957

[AC] [RCL] [S] — S= / 0.3010299957

Important

The following table shows how some variables are used for certain types of calculations. You should not assign other values to these variables when performing the type of calculations noted in the table.

Calculation Type	Variables Used
Differential/Quadratic Differential	F, G, H
Integration	K, L, M, N
Single-variable statistics (SD Mode)	U, V, W
Paired-variable statistics (LR Mode)	P, Q, R, U, V, W

• Independent memory

The "independent memory" lets you directly add to and subtract from variable M with a simple operation. This capability is very useful when you want to perform a series of calculations and accumulate their results for a grand total.

Example **To assign the value 123 to independent memory.**

[AC] [1] [2] [3] [M+] — 123 / 123

Recall the contents of the independent memory.

[AC] [RCL] [M] — M= / 123

Add 25 to memory contents and subtract 12.

[2] [5] [M+] [1] [2] [SHIFT] [M-] — 25 / 25 / 12 / 12

Now you can check memory contents.

[AC] [RCL] [M] — M= / 136

- To clear the independent memory, simply assign a value of zero: [0] [STO] [M]
- Note that the [M+] and [SHIFT] [M-] operations cannot be performed in the SD Mode and LR Mode.

Difference between [STO] [M] and [M+], [SHIFT] [M-]

Since independent memory is actually a variable (M), you can also assign values to it using the variable assignment operation [STO] [M], [M+] and [SHIFT] [M-]. Note, however, that [STO] [M] deletes anything currently stored in independent memory and replaces it with the newly assigned value. The independent memory operation [M+] or [SHIFT] [M-], on the other hand, adds to or subtracts from the value that is currently stored in independent memory.

Example 1 **To use [STO] [M] to assign the value 123 to variable M and then use [STO] [M] to assign the value 456 to variable M.**

[AC] [1] [2] [3] [STO] [M] — 123 / M= / 123

[AC] [4] [5] [6] [STO] [M]

```
456
M=
           456
```

[AC] [RCL] [M]

```
M=
           456
```

Example 2 **To use [STO] [M] to assign the value 123 to variable M and then use [M+] to add the value 456 to independent memory (variable M).**

[AC] [1] [2] [3] [STO] [M]

```
123
M=
           123
```

[AC] [4] [5] [6] [M+]

```
456
           456
```

[AC] [RCL] [M]

```
M=
           579
```

• Arrays

The array capabilities of the calculator lets you use variable names that consist of an alpha character followed by a value (called an "index") inside of square brackets. The following shows some examples of array variable names.

Value Memories	Array Memories
A	A[0] B[–1]
B	A[1] B[0]
C	A[2] B[1]

Arrays help to make programs shorter and simpler.

• Expanding variable memory

You can convert memory normally assigned to program storage to variable memory. Doing so makes it possible to increase the number of variables available from the standard 26 to up to 476. Each additional variable takes up 10 bytes of memory.

Number of variables	26	27	28	·····	476
Memory remaining (bytes)	4500	4490	4480	·····	0

• See page 124 for information on memory requirements for programs.

To expand variable memory

The key operation sequence to expand variable memory is: [SHIFT] [Defm] <number of new variables> [EXE]

Example **To expand variable memory by 10, for a total of 36 variables.**

[SHIFT] [Defm] [1] [0] [EXE]

```
MEMORY :      36
PROGRAM:       0

 4400 Bytes Free
```

- If there is not enough memory to increase the number of variables to the level you want, the above operation produces a Arg ERROR message.
- You can check how much memory is available by inputting: [SHIFT] [Defm] [EXE] .
- You can also include a variable memory expansion operation inside a program using the following syntax: Defm <number of new variables>.

To return variable memory to the standard configuration

The key operation sequence to return the number of available variables to the standard 26 is: [SHIFT] [Defm] [0] [EXE] .

• About memory names

You can use the additional memories you create from program memory just as you use the original 26. The names of the additional memories are Z[1], Z[2], Z[3], etc. If you increase the number of value memories by 5, you can access the original 26 memories, plus memories Z[1] through Z[5].

Example **To assign the value 123 to variable Z [2].**

[SHIFT] [Defm] [2] [EXE]

```
MEMORY :      28
PROGRAM:       0

 4480 Bytes Free
```

[AC] [ALPHA] [Z] [ALPHA] [[] [2] [ALPHA] []]
[ALPHA] [=] [1] [2] [3] [EXE]

```
Z[2]=123
           123
```

Recall the contents of the variable.

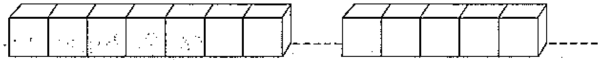
[AC] [ALPHA] [Z] [ALPHA] [[] [2] [ALPHA] []] [EXE]

```
Z[2]
           123
```

• Precautions when using arrays

Array variable names use alpha characters and index numbers, while standard variables use alpha characters only. You should keep in mind, however, that array variables use the same memory as standard variables. Because of this, you must be careful so that the value you assign to one variable does not replace a value already assigned to another variable.

Value memory **Array Memory**



A[0]A[1]A[2]A[3]A[4]A[5]A[6] A[23]A[24]A[25]A[26]A[27]
B[-1]B[0]B[1]B[2]B[3]B[4]B[5] B[22]B[23]B[24]B[25]B[26]
C[-2]C[-1]C[0]C[1]C[2]C[3]C[4] C[21]C[22]C[23]C[24]C[25]
G[-6]G[-5]G[-4]G[-3]G[-2]G[-1]G[0] G[17]G[18]G[19]G[20]G[21]
X[0]X[1]X[2]X[3]X[4]
Y[-1]Y[0]Y[1]Y[2]Y[3]
Z[-2]Z[-1]Z[0]Z[1]Z[2]

1-5 Using Scientific Constants

This calculator features 20 built-in scientific constants that you can recall any time you need them in the COMP, SD, or LR Mode.

1. Press [FUNCTION] to display the function menu.

[FUNCTION]

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG  6.DSP/CLR
```

2. Press [4] (CONST) to recall the first menu of scientific constants.

[4] (CONST)

```
1.mP 2.F  3.a0
4.c  5.h  6.G
7.e  8.me 9.u
0.NA
```

3. Press [▼] to change to the second menu of constants.

[▼]

```
1.k  2.g  3.R
4.ε0 5.μ0 6.μB
7.ħ  8.mn 9.R∞
0.σ
```

4. While either menu is on the display, input the number that corresponds to the scientific constant you want to recall.

• Use [▲] and [▼] to switch between the two scientific constant menus.

Table of Available Constants

• Data is based on ISO Standards (1992) and CODATA Bulletin No. 63 (1986).

Symbol	Quantity	Numerical Value	Unit
m_p	Proton rest mass	1.6726231E–27	kg
F	Faraday constant	96485.309	C/mol
a_0	Bohr radius	5.29177249E–11	m
c	Speed of light in vacuum	299792458	m/s
h	Planck constant	6.6260755E–34	J·s
G	Gravitational constant	6.67259E–11	Nm^2/kg^2
e	Elementary charge	1.60217733E–19	C
m_e	Electron rest mass	9.1093897E–31	kg
u	Atomic mass unit	1.6605402E–27	kg
N_A	Avogadro constant	6.0221367E+23	mol^{-1}
k	Boltzmann constant	1.380658E–23	J/K
g	Gravitational acceleration	9.80665	m/s^2
R	Molar gas constant	8.314510	J/(mol·k)
ε_0	Permittivity of vacuum	8.854187818E–12	F/m
μ_0	Permeability of vacuum	1.256637061E–06	H/m
μ_B	Bohr magneton	9.2740154E–24	$A{\cdot}m^2$
ħ	Converted Planck constant	1.05457266E–34	J·s
m_n	Neutron rest mass	1.6749286E–27	kg
R_∞	Rydberg constant	10973731.53	m^{-1}
σ	Stefan-Boltzmann constant	5.67051E–08	$W/(m^2{\cdot}k^4)$

• Values shown in the table are produced when the calculator is set to Norm 1.
• Scientific constants cannot be used in the BASE-N Mode.

1. **Speed of light in vacuum (c)**
How much energy is produced when a mass of 2 grams is converted completely into energy?

2 [EXP] [(-)] 3 [×] [FUNCTION] [4] (CONST) [4] (c) [x^2] [EXE] **1.797510357E+14**

2. **Planck constant (h)**
How much energy is lost when an atom emits a single photon with a wavelength equivalent to $\lambda = 5.0 \times 10^{-7}$ m?

[FUNCTION] [4] (CONST) [5] (h) [×] [FUNCTION] [4] (CONST) [4] (c)
[÷] 5 [EXP] [(-)] 7 [EXE] **3.972894922E–19**

3. **Gravitational constant (G)**
What is the attractive force between two people weighing 60kg and 80kg who are standing 70cm apart?

[FUNCTION] [4] (CONST) [6] (G) [×] 60 [×] 80 [÷] 0.7 [x^2] [EXE] **6.536414694E–07**

4. **Elementary charge (e), Electron rest mass (me)**
What is the power and acceleration on electrons when 200V are applied to parallel electrodes spaced 3cm apart?

[FUNCTION] [4] (CONST) [7] (e) [×] 200 [÷] 0.03 [EXE] **1.06811822E–15**
[÷] [FUNCTION] [4] (CONST) [8] (me) [EXE] **1.172546411E+15**

5. **Atomic mass unit (u)**
If the mass of a hydrogen atom is 1.00783amu and the mass of its electrons is 1/1800 of that, what is the mass of the nucleus of the hydrogen atom?

[(] 1.00783 [−] 1.00783 [÷] 1800 [)] [×]
[FUNCTION] [4] (CONST) [9] (u) [EXE] **1.672612484E–27**

6. **Avogadro constant (N_A)**
What is the mass of a single molecule of water?

18 [÷] [FUNCTION] [4] (CONST) [0] (N_A) [EXE] **2.988972336E–23**

7. **Boltzmann constant (k)**
What is the average translational motion energy of a single molecule of ideal gas at 0°C?

3 [÷] 2 [×] [FUNCTION] [4] (CONST) [▼] [1] (k) [×] 273 [EXE] **5.65379451E–21**

8. **Gravitational acceleration (g)**
If a small stone dropped into a pond takes 1.5 seconds to hit the surface of the water, how high above the water was the stone dropped?

[FUNCTION] [4] (CONST) [▼] [2] (g) [×] 1.5 [x^2] [÷] 2 [EXE] **11.03248125**

9. **Permittivity of vacuum (ε_0)**
A capacitor is created by two sheets of copper with a surface area of 700cm² spaced 2mm apart. What is the capacitance of the capacitor when it is inserted into oil with a relative conductance value of 5?

[FUNCTION] [4] (CONST) [▼] [4] (ε_0) [×] 5 [×] 700
[EXP] [(-)] 4 [÷] 2 [EXP] [(-)] 3 [EXE] **1.549482868E–09**

10. **Permeability of vacuum (μ_0)**
If two long conductors are placed 1.1 meters apart in a vacuum, what is the force for every two meters when current of 2A and 3A are applied to each conductor in opposite directions?

[FUNCTION] [4] (CONST) [▼] [5] (μ_0) [×] 3 [×] 2 [÷] [SHIFT] [π]
[÷] 1.1 [EXE] **2.181818182E–06**

1-6 Technical Information

This section provides information on the internal workings of the unit.

■ Calculation Priority Sequence

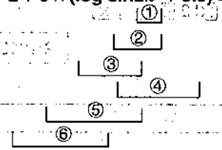
This calculator employs true algebraic logic to calculate the parts of a formula in the following order:

① Coordinate transformation, t-test
Pol (x, y), Rec (r, θ), t(
Differentials, quadratic differentials, integrations, Σ calculations
d/dx, d^2/dx^2, $\int dx$, Σ
② Type A functions
With these functions, the value is entered and then the function key is pressed.
x^2, x^{-1}, $x!$, ° ' ", ENG symbols
③ Power/root
^(x^y), $\sqrt[x]{}$
④ Fractions
a^b/c
⑤ Abbreviated multiplication format in front of π, memory name, or variable name; recursions; scientific constants
2π, 5A, πR, 2mp, etc.
⑥ Type B functions
With these functions, the function key is pressed and then the value is entered.
√, ∛, log, ln, e^x, 10^x, sin, cos, tan, $\sin^{-1}$, $\cos^{-1}$, $\tan^{-1}$, sinh, cosh, tanh, $\sinh^{-1}$, $\cosh^{-1}$, $\tanh^{-1}$, (–), (following in BASE-N Mode only) d, h, b, o, Neg, Not
⑦ Abbreviated multiplication format in front of Type B functions
2√3 , A log2, etc.
⑧ Permutation, combination
nPr, nCr

⑨ ×, ÷
⑩ +, −
⑪ and ⎤
⑫ or, xor, xnor ⎦ BASE-N Mode only.

- When functions with the same priority are used in series, execution is performed from right to left.
  $e^x \ln \sqrt{120} \rightarrow e^x\{\ln(\sqrt{120})\}$
  Otherwise, execution is from left to right.
- Anything contained within parentheses receives highest priority.

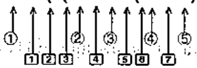
Example 2 + 3 × (log sin2π² + 6.8) = 22.07101691 (angle unit = Rad)



■ Stacks

The unit employs memory blocks, called *stacks*, for storage of low priority values and commands. There is a 10-level *numeric value stack*, a 26-level *command stack*, and a 10-level *program subroutine stack*. If you execute a formula so complex it exceeds the amount of stack space available, an error message appears on the display (Stk ERROR during calculations).

Example 2 × ((3 + 4 × (5 + 4) ÷ 3) ÷ 5) + 8 =



Numeric Value Stack

①	2
②	3
③	4
④	5
⑤	4
⋮	

Command Stack

1	×
2	(
3	(
4	+
5	×
6	(
7	+
⋮	

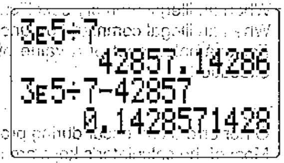
- Calculations are performed according to the priority sequence described on page 41. Once a calculation is executed, it is cleared from the stack.

■ Value Input and Output Limitations

The allowable range for both input and output values is 10 digits for the mantissa and 2 digits for the exponent. Internally, however, the unit performs calculations using 15 digits for the mantissa and 2 digits for the exponent.

Example $3 \times 10^5 \div 7 - 42857 =$

[AC] [3] [EXP] [5] [÷] [7] [EXE]
[3] [EXP] [5] [÷] [7] [−] [4] [2] [8] [5] [7] [EXE]



- Calculation results that are greater than 10^{10} (10 billion) or less than 10^{-2} (0.01) are automatically displayed in exponential form.
- Values are stored in memory with 15 digits for the mantissa and 2 digits for the exponent.

■ Input Capacity

This unit has a 127-byte area for execution of calculations. Each time you press a numeric key or arithmetic operation key, one byte of memory is used. Though such operations as [SHIFT] [x^2] require two key operations, they actually comprise only one function, and therefore, only one byte.

A calculation can consist of up to 127 bytes. Whenever you input the 121st byte of any calculation, the cursor changes from " _ " to " ■ " on the display to let you know that you are running out of memory. If you still need to input more, you should divide your calculation into two or more parts.

Note
- As you input numeric values or commands, they appear flush left on the display. Calculation results, on the other hand, are displayed flush right.

■ Overflow and Errors

Exceeding a specified input or calculation range, or attempting an illegal input causes an error message to appear on the display. Further operation of the calculator is impossible while an error message is displayed. The following events cause an error message to appear on the display.
- When any result, whether intermediate or final, or any value in memory exceeds ±9.999999999 × 10^{99} (Ma ERROR)
- When an attempt is made to perform a function calculation that exceeds the input range (Ma ERROR) (see page 152)
- When an illegal operation is attempted during statistical calculations (Ma ERROR) For example, attempting to obtain $\bar{x}$ or $x\sigma n$ without data input.

- When the capacity of the numeric value stack or command stack is exceeded (Stk ERROR)
  For example, entering 25 successive [(], followed by 2 [+] 3 [×] 4 [EXE].
- When an attempt is made to perform a calculation using an illegal formula (Syn ERROR)
  For example, 5 [×] [×] 3 [EXE].
- When an illegal memory specification is made (Mem ERROR)
- When an illegal command or function argument is used (Arg ERROR)
  For example, input of a value other than 0 to 9 for Fix or Sci during program execution.

Note

- Other errors can occur during program execution. See page 150 for details.
  Most of the calculator's keys are inoperative while an error message is displayed. You can resume operation using the following procedure.
  Press the [AC] key to clear the error and return to normal operation.

■ Exponential Display

During normal calculation, the unit is capable of displaying up to 10 digits. Values that exceed this limit, however, are automatically displayed in exponential format. You can choose between 2 different types of exponential display formats.

Norm 1: 10^{-2}(0.01) > |x|, |x| > 10^{10}

Norm 2: 10^{-9}(0.000000001) > |x|, |x| > 10^{10}

To specify the exponential notation range, first perform the following key operation:
[FUNCTION] [6] (DSP/CLR) [3] (Norm)

Next, press [1] to specify Norm 1 or [2] to specify Norm 2 (page 24).

The current exponential notation range is not indicated by any symbol on the display. You can check which range (Norm 1 or Norm 2) is in effect by performing the following operation.

[AC] 1 [÷] 200 [EXE]

```
1÷200
          5.E-03
```
(Norm 1 display format)

```
1÷200
          0.005
```
(Norm 2 display format)

(All of the examples in this manual show calculation results using Norm 1.)

How to interpret exponential format

```
1.2E12
          1.2E+12
```
Mantissa Exponent

1.2E+12 indicates that the result is equivalent to 1.2×10^{12}. This means that you should move the decimal point in 1.2 twelve places to the right, since the exponent is positive. This results in the value 1,200,000,000,000.

```
1.2E-3
          1.2E-03
```
Mantissa Exponent

1.2E−03 indicates that the result is equivalent to 1.2×10^{-3}. This means that you should move the decimal point in 1.2 three places to the left, since the exponent is negative. This results in the value 0.0012.

■ Calculation Execution Display

When the calculator is busy executing a long, complex calculation or program, a black box (■) flashes in the upper right corner of the display. This black box indicates that the calculator is performing an internal operation.

```
Σ(K²,K,1,10000)■
```

■ Whenever you are having problems...

Whenever your calculations start producing results that are not within expectations, perform the following procedure to get back to standard settings.

1. Press [MODE] [1] to enter the COMP Mode.
2. Press [FUNCTION] [5] (DRG) to display the Unit of Angular Measurement Menu, and then press [1] (Deg) to select degrees.
3. Press [FUNCTION] [6] (DSP/CLR) to display the Display Format/Clear Menu and then press [3] (Norm) [1] to select the Norm 1.
4. Perform your calculation.

If you still experience problems, recheck the content of your calculation, and make sure you are using the correct mode for the type of calculation you are performing.

Chapter 2

Manual Calculations

Chapter 2 Manual Calculations

Manual calculations are those that you input manually, as on the simplest of calculators. They are to be distinguished from programmed calculations. This chapter provides various examples to help you become familiar with the manual calculation capabilities of the unit.

2-1 Basic Calculations

■ Arithmetic Calculations

- Enter arithmetic calculations as they are written, from left to right.
- Use the [(-)] key to input the minus sign before a negative value.
- Calculations are performed internally with a 15-digit mantissa. The display is rounded to a 10-digit mantissa before it is displayed.

Example	Operation	Display
$23 + 4.5 - 53 = -25.5$	23 [+] 4.5 [−] 53 [EXE]	−25.5
$56 \times (-12) \div (-2.5) = 268.8$	56 [×] [(-)] 12 [÷] [(-)] 2.5 [EXE]	268.8
$12369 \times 7532 \times 74103 =$ $6.903680613 \times 10^{12}$ (6903680613000)	12369 [×] 7532 [×] 74103 [EXE]	6.903680613E+12
$(4.5 \times 10^{75}) \times (-2.3 \times 10^{-79})$ $= -1.035 \times 10^{-3}$ (−0.001035)	4.5 [EXP] 75 [×] [(-)] 2.3 [EXP] [(-)] 79 [EXE]	−1.035E−03 (Norm 1)
$(2+3) \times 10^2 = 500$	[(] 2 [+] 3 [)] [×] 1 [EXP] 2 [EXE]	500
	• [(] 2 [+] 3 [)] [EXP] 2 does not produce the correct result. Be sure to enter this calculation as shown.	

- For mixed arithmetic calculations, multiplication and division are given priority over addition and subtraction.

Example	Operation	Display
$3 + 5 \times 6 = 33$	3 [+] 5 [×] 6 [EXE]	33
$7 \times 8 - 4 \times 5 = 36$	7 [×] 8 [−] 4 [×] 5 [EXE]	36
$1 + 2 - \underline{3 \times 4 \div 5} + 6 = 6.6$	1 [+] 2 [−] 3 [×] 4 [÷] 5 [+] 6 [EXE]	6.6

■ Calculations Using Parentheses

Example	Operation	Display
$100 - (2 + 3) \times 4 = 80$	100 [−] [(] 2 [+] 3 [)] [×] 4 [EXE]	80
$2 + 3 \times (4 + 5) = 29$	2 [+] 3 [×] [(] 4 [+] 5 [EXE]	29
• The final closed parentheses (immediately before operation of the [EXE] key) may be omitted, no matter how many are required.		
$(7 - 2) \times (8 + 5) = 65$	[(] 7 [−] 2 [)] [(] 8 [+] 5 [EXE]	65
• A multiplication sign immediately before an open parenthesis may be omitted.		
$10 - \{2 + 7 \times (3 + 6)\} = -55$	10 [−] [(] 2 [+] 7 [(] 3 [+] 6 [EXE]	−55
• In this manual, the multiplication sign is always shown.		
$\frac{2 \times 3 + 4}{5} = (2 \times 3 + 4) \div 5 = 2$	[(] 2 [×] 3 [+] 4 [)] [÷] 5 [EXE]	2
$\frac{6}{4 \times 5} = 0.3$	6 [÷] [(] 4 [×] 5 [)] [EXE]	0.3
• The above is identical to 6 [÷] 4 [÷] 5 [EXE].		

■ Percentage Calculations

Example	Operation	Display
• Percentage 26% of $15.00	15 [×] 26 [SHIFT] [%]	3.9
• Premium 15% increase from $36.20	36.2 [×] 15 [SHIFT] [%] [+]	41.63
• Discount 4% discount from $47.50	47.50 [×] 4 [SHIFT] [%] [−]	45.6
• Ratio 75 is what % of 250?	75 [÷] 250 [SHIFT] [%]	30(%)
• Rate of change 141 is an increase of what % from 120?	141 [−] 120 [SHIFT] [%]	17.5(%)
240 is a decrease of what % from 300?	240 [−] 300 [SHIFT] [%]	−20(%)

2-2 Units of Angular Measurement

- See page 22 for full details on specifying the unit of angular measurement.
- Once you specify a unit of angular measurement, it remains in effect until you specify a different one. The specification is retained even if you switch power off.
- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
Result displayed in degrees. To convert 4.25 rad to degrees.	[FUNCTION] [5] (DRG) [1] (Deg) 4.25 [FUNCTION] [5] (DRG) [5] (r) [EXE]	243.5070629
47.3° + 82.5rad = 4774.20181°	47.3 [+] 82.5 [FUNCTION] [5] (DRG) [5] (r) [EXE]	4774.20181

2-3 Trigonometric and Inverse Trigonometric Functions

- Be sure to set the unit of angular measurement before performing trigonometric function and inverse trigonometric function calculations.
- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
sin 63°52'41" = 0.897859012	[FUNCTION] [5] (DRG) [1] (Deg) [sin] 63 [°'"] 52 [°'"] 41 [°'"] [EXE]	0.897859012
$\cos(\frac{\pi}{3}\text{rad}) = 0.5$	[FUNCTION] [5] (DRG) [2] (Rad) [cos] [(] [SHIFT] [π] [÷] 3 [)] [EXE]	0.5
tan(−35gra) = − 0.6128007881	[FUNCTION] [5] (DRG) [3] (Gra) [tan] [(−)] 35 [EXE]	−0.6128007881
2·sin 45° × cos 65° = 0.5976724775	[FUNCTION] [5] (DRG) [1] (Deg) 2 [×] [sin] 45 [×] [cos] 65 [EXE] (↑ [×] can be omitted.)	0.5976724775

2-4 Logarithmic and Exponential Functions

- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
log 1.23 ($\log_{10}1.23$) = $8.990511144 \times 10^{-2}$	[log] 1.23 [EXE]	0.08990511144
ln90 ($\log_e 90$) = 4.49980967	[ln] 90 [EXE]	4.49980967
$10^{1.23}$ = 16.98243652 (To obtain the antilogarithm of common logarithm 1.23)	[SHIFT] [10^x] 1.23 [EXE]	16.98243652
$e^{4.5}$ = 90.0171313 (To obtain the antilogarithm of natural logarithm 4.5)	[SHIFT] [e^x] 4.5 [EXE]	90.0171313
$10^4 \cdot e^{-4} + 1.2 \cdot 10^{2.3}$ = 422.5878667	[SHIFT] [10^x] 4 [×] [SHIFT] [e^x] [(−)] 4 [+] 1.2 [×] [SHIFT] [10^x] 2.3 [EXE]	422.5878667
$(-3)^4 = (-3) \times (-3) \times (-3) \times (-3) = 81$	[(] [(−)] 3 [)] [^] 4 [EXE]	81

(Continued on the next page)

Example	Operation	Display
$-3^4 = -(3\times3\times3\times3) = -81$	[(-)] 3 [^] 4 [EXE]	−81
$5.6^{2.3} = 52.58143837$	5.6 [^] 2.3 [EXE]	52.58143837
$\sqrt[7]{123}\ (= 123^{\frac{1}{7}}) = 1.988647795$	7 [SHIFT] [$\sqrt[x]{}$] 123 [EXE]	1.988647795

2-5 Hyperbolic and Inverse Hyperbolic Functions

- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
sinh 3.6=18.28545536	[FUNCTION] [1] (MATH) [▼] [▼] [1] (sinh) 3.6 [EXE]	18.28545536
$\cosh^{-1}\left(\frac{20}{15}\right) = 0.7953654612$	[FUNCTION] [1] (MATH) [▼] [▼] [5] ($\cosh^{-1}$) [(] 20 [÷] 15 [)] [EXE]	0.7953654612
Determine the value of x when $\tanh 4x = 0.88$.		
$x = \frac{\tanh^{-1}0.88}{4} = 0.3439419141$	[FUNCTION] [1] (MATH) [▼] [▼] [6] ($\tanh^{-1}$) 0.88 [÷] 4 [EXE]	0.3439419141

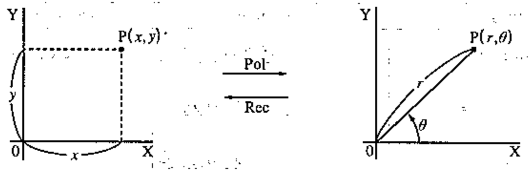
2-6 Other Functions

- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
$\sqrt{2} + \sqrt{5} = 3.65028154$	[√] 2 [+] [√] 5 [EXE]	3.65028154
$(-3)^2 = (-3)\times(-3) = 9$	[(] [(-)] 3 [)] [x^2] [EXE]	9
$-3^2 = -(3\times3) = -9$	[(-)] 3 [x^2] [EXE]	−9
$2^2 + 3^2 + 4^2 + 5^2 = 54$	2 [x^2] [+] 3 [x^2] [+] 4 [x^2] [+] 5 [x^2] [EXE]	54
$\frac{1}{\frac{1}{3} - \frac{1}{4}} = 12$	[(] 3 [SHIFT] [x^{-1}] [−] 4 [SHIFT] [x^{-1}] [)] [SHIFT] [x^{-1}] [EXE]	12
$8!(= 1\times2\times3\times\ldots\times8) = 40320$	8 [FUNCTION] [1] (MATH) [5] ($x!$) [EXE]	40320
$\sqrt[3]{-27} = -3$	[SHIFT] [$\sqrt[3]{}$] [(-)] 27 [EXE]	−3
What is the absolute value of the common logarithm of $\frac{3}{4}$?		
$\left\lvert\log\frac{3}{4}\right\rvert = 0.1249387366$	[FUNCTION] [1] (MATH) [▼] [1] (Abs) [log] [(] 3 [÷] 4 [)] [EXE]	0.1249387366
What is the integer part of −3.5?	[FUNCTION] [1] (MATH) [▼] [2] (Int) [(-)] 3.5 [EXE]	−3
What is the decimal part of −3.5?	[FUNCTION] [1] (MATH) [▼] [3] (Frac) [(-)] 3.5 [EXE]	−0.5
What is the nearest integer not exceeding −3.5?	[FUNCTION] [1] (MATH) [▼] [4] (Intg) [(-)] 3.5 [EXE]	−4

2-7 Coordinate Conversion

- Rectangular Coordinates
- Polar Coordinates



- Calculation results are assigned to variables I and J.

	I	J
Pol	r	θ
Rec	x	y

- With polar coordinates, θ can be calculated and displayed within a range of $-180° < \theta \leqq 180°$ (radians and grads have same range).

- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
To calculate r and $\theta°$ when $x = 14$ and $y = 20.7$	[FUNCTION] [5] (DRG) [1] (Deg) [FUNCTION] [1] (MATH) [▼] [5] (Pol ( ) 14 [,] 20.7 [)] [EXE]	$r =$ **24.98979791** $\theta =$ **55.92839019**
	(Continuing) [ALPHA] [J] [EXE] [SHIFT] [°'"]	**55°55'42.2"**

2-8 Permutation and Combination

- Permutation $nPr = \dfrac{n!}{(n-r)!}$
- Combination $nCr = \dfrac{n!}{r!(n-r)!}$

- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
To calculate the possible number of different arrangements using 4 items selected from among of 10 items. $_{10}P_4 = 5040$	**10** [FUNCTION] [1] (MATH) [7] (nPr) **4** [EXE]	**5040**
To calculate the possible number of different combinations of 4 items that can be selected from among 10 items. $_{10}C_4 = 210$	**10** [FUNCTION] [1] (MATH) [8] (nCr) **4** [EXE]	**210**

2-9 Fractions

- Fractional values are displayed with the integer first, followed by the numerator and then the denominator.
- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
$\frac{2}{5} + 3\frac{1}{4} = 3\frac{13}{20}$ $= 3.65$	2[a b/c]5[+]3[a b/c]1[a b/c]4[EXE] (Conversion to decimal) [a b/c]	3⌟13⌟20 3.65
• Fractions can be converted to decimal values and vice versa.		
$3\frac{456}{78} = 8\frac{11}{13}$ (Reduced)	3[a b/c]456[a b/c]78[EXE] (Continuing) [SHIFT][d/c]	8⌟11⌟13 115⌟13
• Fractions and improper fractions that can be reduced become reduced fractions when you press a calculation command key. Press [SHIFT][d/c] to convert the value to an improper fraction.		
$\frac{1}{2578} + \frac{1}{4572}$ $= 6.066202547 \times 10^{-4}$	1[a b/c]2578[+]1[a b/c]4572[EXE]	6.066202547E- 04 (Norm 1)
• When the total number of characters, including integer, numerator, denominator and deliminator marks exceeds 10, the input fraction is automatically displayed in decimal format.		
$\frac{1}{2} \times 0.5 = 0.25$	1[a b/c]2[×][.]5[EXE]	0.25
• Calculations containing both fractions and decimals are calculated in decimal format.		
$\frac{1}{\frac{1}{3}+\frac{1}{4}} = 1\frac{5}{7}$	1[a b/c][(]1[a b/c]3[+]1[a b/c]4[)][EXE]	1⌟5⌟7
• You can include fractions within the numerator or denominator of a fraction by putting the numerator or denominator in parentheses.		

2-10 Engineering Notation Calculations

Input engineering symbols using the Engineering Notation Menu from the MATH Menu, as described on page 32.

Perform the following operation to change a displayed value to the corresponding Engineering Notation.

[FUNCTION][6] (DSP/CLR) [4] (Eng)

Each time you perform this operation, the display changes between Engineering Notation and standard (non-engineering) notation.

- The unit automatically selects the Engineering Notation that makes the numeric value fall within the range of 1 to 999.
- The following calculations cannot be performed in the BASE-N Mode.

Example	Operation	Display
999k (kilo) + 25k (kilo) = 1.024M (mega)	[FUNCTION][6] (DSP/CLR) [4] (Eng) 999 [FUNCTION][1] (MATH) [▼][▼][▼] [6] (k) [+] 25 [FUNCTION][1] (MATH) [▼][▼][▼][6] (k) [EXE] [FUNCTION][6] (DSP/CLR) [4] (Eng)	 1.024M 1024000
9 ÷ 10 = 0.9 = 900m (milli)	[FUNCTION][6] (DSP/CLR) [4] (Eng) 9[÷]10[EXE]	 900.m
(Converts the displayed value to the next higher engineering unit, by shifting the decimal point three places to the right.)	[SHIFT][ENG] [SHIFT][ENG]	0.9 0.0009k
(Converts the displayed value to the next lower engineering unit, by shifting the decimal point three places to the left.)	[SHIFT][ENG] [SHIFT][ENG] [SHIFT][ENG] [SHIFT][ENG]	0.9 900.m 900000.µ 900.m

2-11 Number of Decimal Places, Number of Significant Digits, Exponential Notation

- See page 23 for details on specifying the number of decimal places.
- See page 23 for details on specifying the number of significant digits.
- See page 24 for details on specifying the exponential notation.

Example	Operation	Display
100 ÷ 6 = 16.66666666...	100 [÷] 6 [EXE]	16.66666667
(4 decimal places)	[FUNCTION] [6] (DSP/CLR) [1] (Fix) [4]	16.6667
(Cancels specification)	[FUNCTION] [6] (DSP/CLR) [3] (Norm) [1]	16.66666667
(5 significant digits)	[FUNCTION] [6] (DSP/CLR) [2] (Sci) [5]	1.6667E+01
(Cancels specification)	[FUNCTION] [6] (DSP/CLR) [3] (Norm) [1]	16.66666667
• Displayed values are rounded off to the place you specify.		
200 ÷ 7 × 14 = 400	200 [÷] 7 [×] 14 [EXE]	400
(3 decimal places)	[FUNCTION] [6] (DSP/CLR) [1] (Fix) [3]	400.000
(Calculation continues using display capacity of 10 digits)		
	200 [÷] 7 [EXE]	28.571
	[×]	Ans × _
	14 [EXE]	400.000
If the same calculation is performed using the specified number of digits:		
	200 [÷] 7 [EXE]	28.571
(The value stored internally is cut off to the number of decimal places you specify.)		
	[SHIFT] [Rnd]	28.571
	[×]	Ans × _
	14 [EXE]	399.994
(Cancels specification)	[FUNCTION] [6] (DSP/CLR) [3] (Norm) [1]	399.994

2-12 Calculations Using Memory

■ Independent Memory

- Values can be directly added to or subtracted from memory. You can view the result of each individual calculation and accumulate a grand total in the memory.

Example	Operation	Display
23 + 9 = 32	23 [+] 9 [STO] [M]	32
53 − 6 = 47	53 [−] 6 [M+]	47
−) 45 × 2 = 90	45 [×] 2 [SHIFT] [M−]	90
99 ÷ 3 = 33	99 [÷] 3 [M+]	33
(Total) 22	[RCL] [M]	22
• Use [STO] [M] to store first value. This clears any previous memory contents. Note that [M+] and [SHIFT] [M−] are used in place of [EXE].		
7 + 7 + 7 +(2 × 3) + (2 × 3) +(2 × 3) − (2 × 3) = 33	7 [STO] [M] [M+] [M+] 2 [×] 3 [M+] [M+] [M+] [SHIFT] [M−] [RCL] [M]	33

■ Variable Memories

The 26 variable memories can be used for storage of data, constants and any other numeric values.

Example	Operation	Display
193.2 ÷ 23 = 8.4	193.2 [STO] [A] [÷] 23 [EXE]	8.4
193.2 ÷ 28 = 6.9	[RCL] [A] [÷] 28 [EXE]	6.9
$\frac{9 \times 6 + 3}{(7-2) \times 8} = 1.425$	9 [×] 6 [+] 3 [STO] [B]	57
	[(] 7 [−] 2 [)] [×] 8 [STO] [C]	40
	[ALPHA] [B] [÷] [ALPHA] [C] [EXE]	1.425
• The same result can be produced by entering [(] 9 [×] 6 [+] 3 [)] [÷] [(] [(] 7 [−] 2 [)] [×] 8 [)] [EXE].		

Chapter 3

Differential, Quadratic Differential, Integration, and Σ Calculations

Chapter 3 Differential, Quadratic Differential, Integration, and Σ Calculations

3-1 Differential Calculations

After you select [2] (d/dx) from the MATH Menu, you can input differentials using following format.

[FUNCTION] [1] (MATH) [2] (d/dx) $f(x)$ [,] a [,] Δx [)]

- Δx — Increase/decrease of x
- a — Point for which you want to determine the derivative

$$d/dx\,(f(x),\,a,\,\Delta x) \Rightarrow \frac{d}{dx}\,f(a)$$

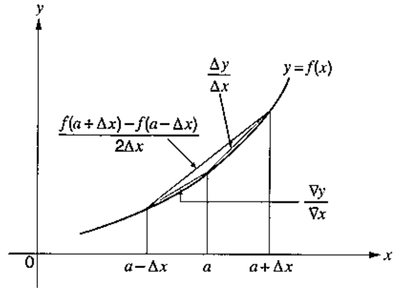
The differentiation for this type of calculation is defined as:

$$f'(a) = \lim_{\Delta x \to 0} \frac{f(a+\Delta x) - f(a)}{\Delta x}$$

In this definition, *infinitesimal* is replaced by a *sufficiently small* Δx, with the value in the neighborhood of $f'(a)$ calculated as:

$$f'(a) = \frac{f(a+\Delta x) - f(a)}{\Delta x}$$

In order to provide the best precision possible, this unit employs central difference to perform differential calculations. The following illustrates central difference.



The slopes of point a and point $a + \Delta x$, and of point a and point $a - \Delta x$ in function $y = f(x)$ are as follows:

$$\frac{f(a+\Delta x) - f(a)}{\Delta x} = \frac{\Delta y}{\Delta x}, \quad \frac{f(a) - f(a-\Delta x)}{\Delta x} = \frac{\nabla y}{\nabla x}$$

In the above, $\Delta y/\Delta x$ is called the forward difference, while $\nabla y/\nabla x$ is the backward difference. To calculate derivatives, the unit takes the average between the value of $\Delta y/\Delta x$ and $\nabla y/\nabla x$, thereby providing higher precision for derivatives.

This average, which is called the *central difference*, is expressed as:

$$f(a) = \frac{1}{2}\left(\frac{f(a+\Delta x) - f(a)}{\Delta x} + \frac{f(a) - f(a-\Delta x)}{\Delta x}\right)$$

$$= \frac{f(a+\Delta x) - f(a-\Delta x)}{2\Delta x}$$

■ To Perform a Differential Calculation

Example **To determine the derivative at point $x = 3$ for the function $y = x^3 + 4x^2 + x - 6$, when the increase/decrease of x is defined as $\Delta x = 1\text{E}-5$.**

Input the function $f(x)$.

[AC] [FUNCTION] [1] (MATH) [2] (d/dx)
[ALPHA] [X] [^] [3] [+] [4] [ALPHA] [X] [x^2]
[+] [ALPHA] [X] [−] [6] [,]

```
d/dx(X^3+4X²+X-6
,_
```

Input point $x = a$ for which you want to determine the derivative.

[3] [,]

```
d/dx(X^3+4X²+X-6
,3,_
```

Input Δx, which is the increase/decrease of x.

[1] [EXP] [(−)] [5] [)]

```
d/dx(X^3+4X²+X-6
,3,1E-5)_
```

[EXE]

```
d/dx(X^3+4X²+X-6
,3,1E-5)
              52
```

- X is the only expression that can be used in the function $f(x)$. If you use any other variable name (A through Z), that variable name is regarded as a constant, using the value currently assigned to the variable in the calculation.
- Input of Δx for the increase/decrease of x can be skipped. When you do, the unit automatically uses a value for Δx that is appropriate for the value of $x = a$, which you specified as the point for which you wanted to determine the derivative.
- In general, calculation precision is ±1 at the least significant digit of the result.

■ Applications of Differential Calculations

- Differentials can be added, subtracted, multiplied and divided with each other.

Example $\frac{d}{dx}f(a) = f'(a)$, $\frac{d}{dx}g(a) = g'(a)$

Therefore:

$f'(a) + g'(a)$, $f'(a) \times g'(a)$

- Differential results can be used in addition, subtraction, multiplication, and division, and in functions.

Example $2 \times f'(a)$, $\log(f'(a))$

- Functions can be used in any of the terms ($f(x)$, a, Δx) of a differential.

Example $\frac{d}{dx}(\sin x + \cos x, \sin 0.5)$

- Note that you cannot use differential, integration, or Σ calculations inside of a differential calculation term.

Important

- Pressing [AC] during calculation of a differential (while the cursor is not shown on the display) interrupts the calculation.
- Always perform trigonometric integrations using radians (Rad Mode) as the unit of angular measurement.
- Differential calculations use variables F through H for storage, deleting any contents that were previously stored. This also means that you cannot use these variables during differential calculations.

Variable	F	G	H
Data Stored	a	Δx	$f'(a)$

In addition to the above, the value for derivative a is assigned to variable X.

3-2 Quadratic Differential Calculations

After you select [3] (d^2/dx^2) from the MATH Menu, you can input quadratic differentials using either of the two following formats.

[FUNCTION] [1] (MATH) [3] (d^2/dx^2) $f(x)$ [,] a [,] n [)]

- n: Final boundary (n = 1 to 15) — Input of a value for n can be skipped.
- a: Differential coefficient point

$$d^2/dx^2\,(f(x),\,a,\,n) \Rightarrow \frac{d^2}{dx^2}f(a)$$

Quadratic differential calculations produce an approximate differential value using the following second order differential formula, which is based on Newton's polynomial interpretation.

$$f''(x) = \{-f(x-2h) + 16f(x-h) - 30f(x) + 16f(x+h) - f(x+2h)\}\,/\,(12h^2)$$

In this expression, values for *sufficiently small increments of* x are sequentially calculated using the following formula, with the value of m being substituted as m = 1, 2, 3 and so on.

$$h = 1/5^m$$

The calculation is finished when the value of $f''(x)$ based on the value of h calculated using the last value of m, and the value of $f''(x)$ based on the value of h calculated using the current value of m are identical by the time the upper limit n is reached.

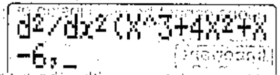
- Normally, you should not input a value for n. This automatically assigns a default value of 7 for n. It is recommended that you only input a value for n when required for calculation precision.
- Inputting a larger value for n does not necessarily produce greater precision.

■ To Perform a Quadratic Differential Calculation

Example **To determine the quadratic differential coefficient at the point where $x = 3$ for the function $y = x^3 + 4x^2 + x - 6$. In this case, input 6 as n, which is final boundary.**

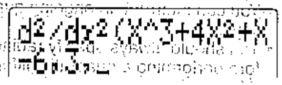
Input $f(x)$.

[AC] [FUNCTION] [1] (MATH) [3] (d^2/dx^2)
[ALPHA] [X] [^] [3] [+] [4] [ALPHA] [X] [x^2]
[+] [ALPHA] [X] [−] [6] [,]



Input 3 as point a, which is differential coefficient point.

[3] [,]



Input 6 as n, which is final boundary.

[6] [)]

```
d²/dx²(X^3+4X²+X
-6,3,6)_
```

[EXE]

```
d²/dx²(X^3+4X²+X
-6,3,6)
                26
```

- In the function $f(x)$, only X can be used as a variable in expressions. Other variables are treated as constants, and the value currently assigned to that variable is applied during the calculation.
- Input of the closing parenthesis following the final boundary value can be omitted.
- In general, calculation precision is ±1 at the least significant digit of the result.

■ Applications of Quadratic Differential Calculations

- Arithmetic operations can be performed using two quadratic differentials.

Example $\frac{d^2}{dx^2}f(a) = f''(a)$, $\frac{d^2}{dx^2}g(a) = g''(a)$

Therefore:

$f''(a) + g''(a)$, $f''(a) \times g''(a)$

- The result of a quadratic differential calculation can be used in a subsequent arithmetic or function calculation.

Example $2 \times f''(a)$, $\log(f''(a))$

- Functions can be used within the terms ($f(x)$, a, n) of a quadratic differential expression.

Example $\frac{d^2}{dx^2}(\sin x + \cos x, \sin 0.5)$

- Note that differential, quadratic differential, integration, and Σ calculation expressions cannot be used inside of the terms of a quadratic differential expression.

Important

- Use only integers within the range of 1 to 15 for the value of final boundary n. Use of a value outside this range produces an Ma ERROR.
- You can interrupt an ongoing quadratic differential calculation by pressing the [AC] key.
- You should always specify radians (Rad) as the unit of angular measurement before performing a quadratic differential calculation using trigonometric functions.

- Variables F, G, and H are used by the calculator during quadratic differential calculations. You can recall the values currently assigned to these variables at any time to check the details of the calculation. Also remember that you should never use these three variables for other purposes when performing quadratic differential calculations.

Variable	F	G	H
Data	a	n	$f''(a)$

In addition to the above, differential coefficient a is assigned to variable X after execution of a quadratic differential calculation.

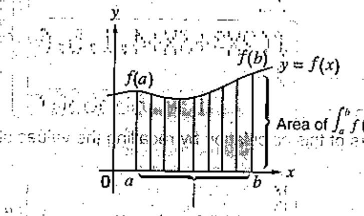
3-3 Integration Calculations

After you select [1] ($\int dx$) from the MATH Menu, you can input integrations using following format.

[FUNCTION] [1] (MATH) [1] ($\int dx$) $f(x)$ [,] a [,] b [,] n [)]

- n: Number of Divisions (value for n in $N = 2^n$, n is an integer from 1 through 9)
- b: End Point
- a: Start Point

$\int(f(x), a, b, n) \Rightarrow \int_a^b f(x)dx, N = 2^n$



Integration calculations are performed by applying Simpson's Rule for the $f(x)$ function you input. This method requires that the number divisions be defined as $N = 2^n$, where the value of n is an integer in the range of 1 through 9. If you do not specify a value for n, the calculator automatically assigns a value in accordance with the integration being performed.

As shown in the illustration above, integration calculations are performed by calculating integral values from a through b for the function $y = f(x)$ where $a \leq x \leq b$, and $f(x) \geqq 0$*. This in effect calculates the surface area of the shaded area in the illustration.

* If $f(x) < 0$ where $a \leq x \leq b$, the surface area calculation produces negative values (surface area × – 1).

Also note that the calculator uses the following variables to store data during integration calculations:

Variable	K	L	M	N
Data Stored	a	b	$N = 2^n$	$\int_a^b f(x)\,dx$

■ To Perform an Integration Calculation

Example **To perform the integration calculation for the function $\int_1^5 (2x^2 + 3x + 4)\,dx$**

Input the function $f(x)$.

[AC] [FUNCTION] [1] (MATH) [1] (∫dx) [2] [ALPHA] [X] [x²] [+] [3] [ALPHA] [X] [+] [4] [,]

```
∫(2X²+3X+4,_
```

Input the start point and end point.

[1] [,] [5] [,]

```
∫(2X²+3X+4,1,5,_
```

Input the number of divisions.

[6] [)]

```
∫(2X²+3X+4,1,5,6
)_
```

[EXE].

```
∫(2X²+3X+4,1,5,6
)
          134.6666667
```

The result takes a few seconds to appear on the display.

You can confirm the parameters of this calculation by recalling the values stored in the value memories.

[ALPHA] [K] [EXE]

```
K
                    1
```

a

[ALPHA] [L] [EXE]

```
L
                    5
```

b

[ALPHA] [M] [EXE]

```
M
                   64
```

N

[ALPHA] [N] [EXE]

```
N
          134.6666667
```

$\int_a^b f(x)dx$

- X is the only expression that can be used in the function $f(x)$. If you use any other variable name (A through Z), that variable name is regarded as a constant, and the value currently assigned to the variable is used in the calculation.
- n and parentheses may be omitted. If you omit n, the calculator automatically selects the most appropriate value.
- In general, calculation precision is ±1 at the least significant digit of the result.

■ Applications of Integration Calculation

- Integrals can be used in addition, subtraction, multiplication and division.

Example $\int_a^b f(x)\,dx + \int_c^d g(x)\,dx$

- Integration results can be used in addition, subtraction, multiplication and division, in functions.

Example $2 \times \int_a^b f(x)\,dx$, $\log(\int_a^b f(x)\,dx)$

- Functions can be used in any of the terms ($f(x)$, a, b, n) of an integral.

Example $\int_{\sin 0.5}^{\cos 0.5} (\sin x + \cos x)\,dx$
$= \int(\sin x + \cos x, \sin 0.5, \cos 0.5, 5)$

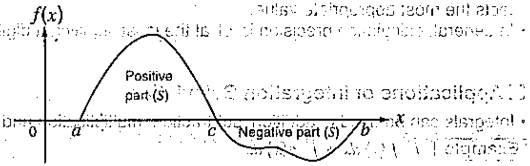
- Note that you cannot use differential, integration, or Σ calculations inside of an integration calculation term.

Important

- Pressing [AC] during calculation of an integral (while the cursor is not shown on the display) interrupts the calculation.
- Always perform trigonometric integrations using radians (Rad Mode) as the unit of angular measurement.
- Integration calculations use variables K through N for storage, deleting any contents that may be already stored. This also means that you cannot use these variables during integration calculations.
  In addition to the above, the value that represents division beginning point a is stored in variable X following completion of the integration calculation.
- This unit utilizes Simpson's rule for integration calculation. As the number of significant digits is increased, more calculation time is required. In some cases, calculation results may be erroneous even after considerable time is spent performing a calculation. In particular, when significant digits are less than 1, an ERROR (Ma ERROR) sometimes occurs.
- Integration involving certain types of functions or ranges can result in relatively large errors being generated in the values produced.

- Note the following points to ensure correct integration values.

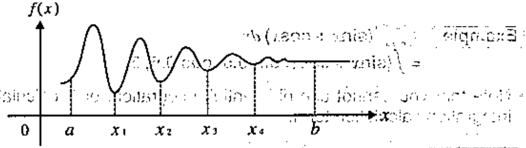
(1) When cyclical functions for integration values become positive or negative for different divisions, perform the calculation for single cycles, or divide between negative and positive, and then add the results together.



$$\int_a^b f(x)dx = \int_a^c f(x)dx + \left(-\int_c^b f(x)dx\right)$$

Positive part (S) Negative part (S)

(2) When minute fluctuations in integration divisions produce large fluctuations in integration values, calculate the integration divisions separately (divide the large fluctuation areas into smaller divisions), and then add the results together.



$$\int_a^b f(x)dx = \int_a^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \dots + \int_{x_4}^{b} f(x)dx$$

3-4 Σ Calculations

After you select [4] (Σ( ) from the MATH Menu, you can input the following Σ calculation format.

[FUNCTION] [4] (MATH) [4] (Σ( ) a_k [,] k [,] α [,] β [)]

- Last term of sequence $\{a_k\}$
- Initial term of sequence $\{a_k\}$
- Variable used by sequence $\{a_k\}$

$$\Sigma(a_k, k, \alpha, \beta) \Rightarrow \sum_{k=\alpha}^{\beta} a_k$$

Σ calculation is the calculation of the partial sum of sequence $\{a_k\}$, using the following formula.

$$S = a_\alpha + a_{\alpha+1} + \dots\dots + a_\beta = \sum_{k=\alpha}^{\beta} a_k$$

■ Example Σ Calculation

Example To calculate the following:

$$\sum_{k=2}^{6}(K^2 - 3K + 5)$$

Input sequence $\{a_k\}$.

[AC] [FUNCTION] [1] (MATH) [4] (Σ( )
[ALPHA] [K] [x^2] [−] [3] [ALPHA] [K]
[+] [5] [,]

Σ(K²-3K+5,_

Input variable used by sequence $\{a_k\}$.

[ALPHA] [K] [,]

Σ(K²-3K+5,K,_

Input the initial term of sequence $\{a_k\}$ and last term of sequence $\{a_k\}$.

[2] [,] [6] [)]

Σ(K²-3K+5,K,2,6)
_

[EXE]

Σ(K²-3K+5,K,2,6)
55

- You can use only one variable in the function for input sequence $\{a_k\}$.
- Input integers only for the initial term of sequence $\{a_k\}$ and last term of sequence $\{a_k\}$.
- Closing parentheses may be omitted.

■ Σ Calculation Applications

- **Arithmetic operations using Σ calculation expressions**

Expressions: $S_n = \sum_{k=1}^{n} a_k$, $T_n = \sum_{k=1}^{n} b_k$

Possible operations: $S_n + T_n$, $S_n - T_n$, etc.

- **Arithmetic and function operations using Σ calculation results**
  $2 \times S_n$, log (S_n), etc.

- **Function operations using Σ calculation terms (a_k, α, β)**
  $\Sigma(x^2 + x, x, 2^2, 5^2 + 1)$, etc.

- Note that you cannot use differential, integration, or Σ calculations inside of a Σ calculation term.

■ Σ Calculation Precautions

- Make sure that the value used as the final term β is greater than the value used as the initial term α. Otherwise, an Ma ERROR will occur.
- To interrupt an ongoing Σ calculation (indicated when the cursor is not on the display), press the AC key.

Chapter 4

Complex Numbers

Chapter 4

Complex Numbers

This calculator is capable of performing the following operations using complex numbers.

- Arithmetic operations (addition, subtraction, multiplication, division)
- Calculation of the reciprocal, square root, and square of a complex number
- Calculation of the absolute value and argument of a complex number
- Calculation of conjugate complex numbers
- Extraction of the real number part
- Extraction of the imaginary number part

4-1 Before Beginning a Complex Number Calculation

Use the following procedure to display the Complex Number Calculation (COMPLX) Menu.

FUNCTION 2 (COMPLX)

```
1.Abs    2.Arg
3.Conjg  4.ReP
5.ImP
```

"1. Abs" .... Absolute value of a complex number
"2. Arg" .... Argument of a complex number
"3. Conjg" .... Conjugate complex number
"4. ReP" .... Real part of complex number
"5. ImP" .... Imaginary part of complex number

4-2 Performing Complex Number Calculations

The following examples show how to perform each of the complex number calculations available with this calculator.

■ Arithmetic Operations

Arithmetic operations are the same as those you use for manual calculations (page 48). You can even use parentheses and memory.

Example 1 $(1 + 2i) + (2 + 3i) =$

AC (1 + 2 i) +
(2 + 3 i) EXE

```
(1+2i)+(2+3i)
           3+5i
```

Example 2 $(2 + i) \times (2 - i) =$

AC (2 + i) ×
(2 − i) EXE

```
(2+i)×(2-i)
              5
```

■ Reciprocals, Square Roots, and Squares

Example $\sqrt{(3 + i)} =$

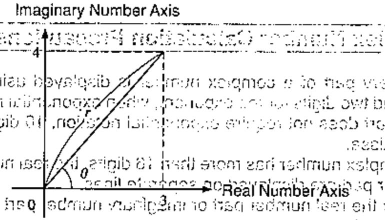
AC √ (3 + i) EXE

```
√(3+i)
   1.755317302
 +0.2848487846i
```

■ Absolute Value and Argument

The unit regards a complex number in the format Z = a + bi as a coordinate on of a Gaussian plane, and calculates absolute value |Z| and argument (arg).

Example **To calculate absolute value (*r*) and argument (*θ*) for the complex number 3+4*i*, with the unit of angular measurement set for degrees.**



AC FUNCTION 2 (COMPLX) 1 (Abs)
(3 + 4 i) EXE
(Calculation of absolute value)

```
Abs (3+4i)
              5
```

AC FUNCTION 2 (COMPLX) 2 (Arg)
(3 + 4 i) EXE
(Calculation of argument)

```
Arg (3+4i)
    53.13010235
```

- The result of the argument calculation differs in accordance with the current unit of angular measurement setting (degrees, radians, grads).

■ Conjugate Complex Numbers

A complex number of the format $a + bi$ becomes a conjugate complex number of the format $a - bi$.

Example **To calculate the conjugate complex number for the complex number 2 + 4*i*.**

[AC] [FUNCTION] [2] (COMPLX)
[3] (Conjg) [(] [2] [+] [4] [i] [)] [EXE]

```
Conjg (2+4i)
          2-4i
```

■ Extraction of Real and Imaginary Number Parts

Use the following procedure to extract real part a and imaginary part b from a complex number with the format $a + bi$.

Example **To extract the real and imaginary parts of the complex number 2 + 5*i*.**

[AC] [FUNCTION] [2] (COMPLX) [4] (ReP)
[(] [2] [+] [5] [i] [)] [EXE]
(Real part extraction)

```
ReP (2+5i)
             2
```

[AC] [FUNCTION] [2] (COMPLX) [5] (ImP)
[(] [2] [+] [5] [i] [)] [EXE]
(Imaginary part extraction)

```
ImP (2+5i)
             5
```

4-3 Complex Number Calculation Precautions

- The imaginary part of a complex number is displayed using nine digits for the mantissa and two digits for the exponent, when exponential notation is used. If the imaginary part does not require exponential notation, 10 digits become available for the mantissa.
- When a complex number has more than 16 digits, the real number part and imaginary number part are displayed on separate lines.
- When either the real number part or imaginary number part equals zero, that part is not displayed.
- 10 bytes of memory are used to store the imaginary part whenever you assign a comlex number to a variable (page 33).
- The following functions can be used with complex numbers.
  $\sqrt{\ }$, x^2, x^{-1}
  Int, Frac, Rnd, Intg, Fix, Sci, ENG, $\overleftarrow{\text{ENG}}$, ° ' ", $\overleftarrow{\text{° ' "}}$, a^b/c, d/c

Chapter 5

Recursion Calculations

Chapter 5 Recursion Calculations

This calculator can calculate the following two types of recursions.

- a_n type recursion
  General term of sequence $\{a_n\}$, consisting of a_n and n.
- a_{n+1} type recursion
  Linear two-term recursion, consisting of a_{n+1}, a_n, and n.

5-1 Before Beginning a Recursion Calculation

1. Press [MODE] [6] (a_n) to enter the Recursion Calculation Mode.

[MODE] [6] (a_n)

```
an=_
```

This display appears when a_n type recursion is currently selected. See step 3 below for information on selecting the recursion type.

- Any recursion formula you previously input will also appear on the display when you enter the Recursion Calculation Mode.

2. Press [FUNCTION] to display the function menu.

- The function menu that appears depends on whether a_n type or a_{n+1} type recursion is currently selected.

When a_n type is selected

```
1.n    2.MATH
3.TYPE
```

"1. n" .... Inputs n.
"2. MATH" .... MATH Menu (page 30)
"3. TYPE" .... Specification of recursion type.

When a_{n+1} type is selected

```
1.n    2.an
3.MATH 4.TYPE
```

"1. n" .... Inputs n.
"2. a_n" .... Inputs a_n.
"3. MATH" .... MATH Menu (page 30)
"4. TYPE" .... Specification of recursion type.

3. Display the TYPE menu to select the type of recursion you want.

```
<SELECT an TYPE>
1.an=An+B
2.an+1=Aan+Bn+C
```

"1. a_n = An + B" .... a_n type recursion
"2. a_{n+1} = Aa_n + Bn+C" ..... a_{n+1} type recursion

- a_n = An + B in the above menu represents common term a_n = A × n + B of $\{a_n\}$.

[EXE]

5-2 Performing Recursion Calculations

Example 1 **Input $a_n = n + 2$ and calculate values for a_n and Σa_n as variable n changes in value from 2 through 6.**

1. Select recursion type a_n.

[1]

```
an=_
```

2. Input the formula.

[FUNCTION]

```
1.n    2.MATH
3.TYPE
```

[1] (n) [+] [2]

```
an=n+2_
```

[EXE]

3. Press [EXE].

[EXE]

```
**TABLE Range**
n
Start?
                1
```

- The table range input screen appears, showing the current table range start value.
- The calculator will start from the value you specify and increment by one for each successive calculation.

4. Since we want to use a start value of 2, press [2] to change the current value.

[2]

```
**TABLE Range**
n
Start?
2_
```

5. Press [EXE] to perform the calculation. Each time you press [EXE], the value of n is incremented and the corresponding results for a_n and Σa_n appear on the display.

[EXE]

```
an=n+2
n=        2
an=       4
Σan=      7
```

[EXE]

```
an=n+2
n=        3
an=       5
Σan=     12
```

⋮

[EXE]

```
an=n+2
n=        6
an=       8
Σan=     33
```

- You can continue pressing [EXE] until n = 9,999,999,999.
- To quit the calculation, press [EXIT]. This returns you to the display in step 2.

Note

- Non-linear exponential expressions (ex. $a_n = 2^n - 1$), fractional expressions (ex. $a_n = (n+1)/n$), irrational expressions (ex. $a_n = \sqrt{n} - \sqrt{(n-1)}$), or trigonometric expressions (a_n = sin $2n\pi$) can be input into the general term of $\{a_n\}$ for generation of a numeric table.

Example 2 Input $a_{n+1} = a_n + 5$ and calculate values for a_n and Σa_n as variable n changes in value from 1 through 5. Note that $a_1 = 2$.

1. Select recursion type a_{n+1}.

[2]

```
an+1=_
```

2. Input the formula.

[FUNCTION]

```
1.n      2.an
3.MATH   4.TYPE
```

[2] (a_n) [+] [5]

```
an+1=an+5_
```

3. Press [EXE].

[EXE]

```
**TABLE Range**

a1 ?
              1
```

- The table range input screen appears, showing the current value for a_1, which is the first term of sequence $\{a_n\}$.

4. Input the value you want to use for a_1.

[2]

```
**TABLE Range**

a1 ?
2_
```

5. Press [EXE].

[EXE]

```
**TABLE Range**
n
Start?
              1
```

- Another table range input screen appears, showing the current table range start value.
- The calculator will start from the value you specify and increment by one for each successive calculation.
- Since we want to start from 1, which is the value shown on the display, we do not need to input anything to change it.

6. Press [EXE] to perform the calculation. Each time you press [EXE], the value of n is incremented and the corresponding results for a_n and Σa_n appear on the display.

[EXE]

```
an+1=an+5
n=         1
an=        2
Σan=       2
```

[EXE]

```
an+1=an+5
n=         2
an=        7
Σan=       9
```

[EXE]

```
an+1=an+5
n=         5
an=       22
Σan=      60
```

- You can continue pressing [EXE] until n = 9,999,999,999.
- To quit the calculation, press [EXIT]. This returns you to the display in step 2.

Important

- If you specify a negative value for the table start value, the sign is dropped.
- If you specify a decimal or fractional value for the table start value, only the integer part is used.
- Specifying a relatively large table start value can cause the calculation to require considerable time to complete.
- A result that is out of the range of the calculator causes an Ma ERROR.
- If you change the unit of angular measurement while the result of a recursion calculation that contains trigonometric functions is on the display, the displayed result is not converted to the new unit. To recalculate using a different unit of angular measurement, go to the COMP Mode and change to the unit you want to use, press [MODE] [6] (a_n), and then perform the recursion calculation again.

Chapter 6

BASE-N Mode Calculations

Chapter 6 BASE-N Mode Calculations

You can use the **BASE-N Mode** to perform calculations with binary, octal, decimal and hexadecimal values. You should also use this mode to convert between number systems and for logical operations.

- You cannot use scientific functions in the BASE-N Mode.
- You can use only integers in the BASE-N Mode, so fractional values are not allowed. If you input a value that includes a decimal part, the unit automatically cuts off the decimal.
- If you attempt to enter a value that is invalid in the number system (binary, octal, decimal, hexadecimal) you are using, the calculator displays an error message. The following shows the numerals that can be used in each number system.
  - Binary: 0, 1
  - Octal: 0, 1, 2, 3, 4, 5, 6, 7
  - Decimal: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  - Hexadecimal: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- The alphabetic characters used in the hexadecimal number appear differently on the display to distinguish them from text characters.
  - Normal Text: A, B, C, D, E, F
  - Hexadecimal Values: A, B, C, D, E, F
- Negative binary, octal, and hexadecimal values are produced using the two's complement of the original value.
- The following are the display capacities for each of the number systems.

Number System	Display Capacity
Binary	16 digits
Octal	11 digits
Decimal	10 digits
Hexadecimal	8 digits

- The following are the calculation ranges for each of the number systems in the BASE-N Mode.

Binary Values
- Positive: $0 \leqq x \leqq 01111111111111111111111111111111$
- Negative: $10000000000000000000000000000000 \leqq x \leqq 11111111111111111111111111111111$

Octal Values
- Positive: $0 \leqq x \leqq 17777777777$
- Negative: $20000000000 \leqq x \leqq 37777777777$

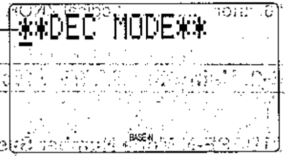
Decimal Values
- Positive: $0 \leqq x \leqq 2147483647$
- Negative: $-2147483648 \leqq x \leqq -1$

Hexadecimal Values
- Positive: $0 \leqq x \leqq$ 7FFFFFFF
- Negative: 80000000 $\leqq x \leqq$ FFFFFFFF

6-1 Before Beginning a Binary, Octal, Decimal, or Hexadecimal Calculation

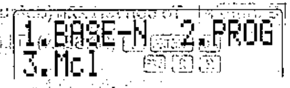
1. Enter the BASE-N Mode.

   MODE 2 (BASE-N)



2. Display the function menu.

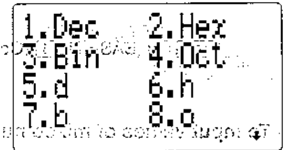
   FUNCTION



"1. BASE-N" .... BASE-N Menu
"2. PROG" .... Program Command Menu (page 132)
"3. Mcl" .... All memory clear (page 24)

3. Display the BASE-N Menu.

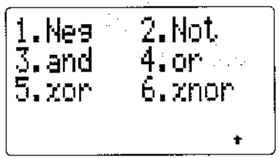
   1 (BASE-N)



"1. Dec" .... Specifies decimal as default
"2. Hex" .... Specifies hexadecimal as default
"3. Bin" .... Specifies binary as default
"4. Oct" .... Specifies octal as default
"5. d" .... Specifies decimal for input value
"6. h" .... Specifies hexadecimal for input value
"7. b" .... Specifies binary for input value
"8. o" .... Specifies octal for input value

4. Press ▼ to change to the second BASE-N Menu.

   ▼



"1. Neg" .... Negation
"2. Not" .... Logical NOT
"3. and" .... Logical AND
"4. or" .... Logical OR
"5. xor" .... Logical XOR
"6. xnor" .... Logical XNOR

6-2 Using the BASE-N Mode

■ BASE-N Mode Number System

• To convert a displayed value from one number system to another

Example To convert 22_{10} (default number system) to its binary or octal value

[AC] [FUNCTION] [1] (BASE-N) [1] (Dec)
[2] [2] [EXE]

```
22
                22
```

[FUNCTION] [1] (BASE-N) [3] (Bin)

```
22
0000000000000000
0000000000010110
```

[FUNCTION] [1] (BASE-N) [4] (Oct)

```
22
     00000000026
```

• To input values of mixed number systems

Example To input 123_{10} or 1010_2, when the default number system is hexadecimal

[AC] [FUNCTION] [1] (BASE-N) [2] (Hex)
[FUNCTION] [1] (BASE-N) [5] (d)
[1] [2] [3] [EXE]

```
d123
        0000007B
```

[FUNCTION] [1] (BASE-N) [7] (b)
[1] [0] [1] [0] [EXE]

```
b1010
        0000000A
```


6-3 BASE-N Mode Calculations

■ Arithmetic Operations

Example 1 To calculate $10111_2 + 11010_2$

[AC] [FUNCTION] [1] (BASE-N) [3] (Bin)
[1] [0] [1] [1] [1] [+]
[1] [1] [0] [1] [0] [EXE]

```
10111+11010
0000000000000000
0000000000110001
```

Example 2 To input and execute $123_8 \times ABC_{16}$, when the default number system is decimal or hexadecimal

[AC] [FUNCTION] [1] (BASE-N) [1] (Dec)
[FUNCTION] [1] (BASE-N) [8] (o)
[1] [2] [3] [×]
[FUNCTION] [1] (BASE-N) [6] (h)
[A] [B] [C] [EXE]

```
o123×hABC
          228084
```

[FUNCTION] [1] (BASE-N) [2] (Hex)

```
o123×hABC
        00037AF4
```

■ Negative Values

Example To calculate the negative of 110010_2

[AC] [FUNCTION] [1] (BASE-N) [3] (Bin)
[FUNCTION] [1] (BASE-N) [▼]
[1] (Neg) [1] [1] [0] [0] [1] [0] [EXE]

```
Neg 110010
1111111111111111
1111111111001110
```

■ Logical Operations

Example 1 To input and execute 120_{16} and AD_{16}

[AC] [FUNCTION] [1] (BASE-N) [2] (Hex)
[1] [2] [0] [FUNCTION] [1] (BASE-N) [▼]
[3] (and) [A] [D] [EXE]

```
120andAD
        00000020
```

Example 2 To calculate 36_8 or 1110_2 to its octal value

[AC] [FUNCTION] [1] (BASE-N) [4] (Oct)
[3] [6] [FUNCTION] [1] (BASE-N) [▼]
[4] (or) [FUNCTION] [1] (BASE-N) [7] (b)
[1] [1] [1] [0] [EXE].

```
36orb1110
        00000000036
```

Example 3 To negate $2FFFED_{16}$

[AC] [FUNCTION] [1] (BASE-N) [2] (Hex)
[FUNCTION] [1] (BASE-N) [▼] [2] (not)
[2] [F] [F] [F] [E] [D] [EXE].

```
Not 2FFFED
          FFD00012
```


Chapter 7

Statistical Calculations

Chapter 7 Statistical Calculations

This calculator can perform both single-variable statistical calculations in the SD Mode using standard deviation, and paired-variable statistical calculations in the LR Mode using regression.

7-1 Single Variable Statistical Calculations

In the **SD Mode** you can obtain population standard deviation, sample standard deviation, the mean of data, sum of the squares of the data, the sum of the data, and the number of data items.

• To input single-variable data

1. First perform the following operation to clear statistical memory, which consists of variables U, V, and W.

   [FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]

- Always make sure you clear statistical memory before performing any statistical calculation.

2. Input the data using the following syntax:

   <data value> [DT] (= [M+])

Example **To input the data 10, 20**

Key operation: 10 [DT] 20 [DT]

- You can input repeated data by simply pressing [DT] again.

Example **To input the data 10, 10**

Key operation: 10 [DT] [DT]

- You can input multiple identical data by specifying the number of repeats.

Example **To input the data 20, 20, 20, 20, 20**

Key operation: 20 [SHIFT] [;] 5 [DT]

(20 — Data; 5 — Number of repeats)

• Deleting single-variable data

The method you use to delete data depends on whether or not you have already stored the data by pressing the [DT] key.

To delete data that is not stored

If you have input data on the keyboard but have not yet stored it by pressing [DT], simply press [AC].

To delete data that is already stored

Use the following syntax to delete data that you have already stored by pressing [DT].

<data to be deleted> [SHIFT] [CL]

Example **To delete the data 10 and 20**

Key operation: 10 [SHIFT] [CL] 20 [SHIFT] [CL]

- You can delete multiple identical data by specifying the number of data items.

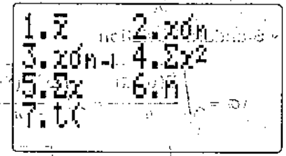
Example **To delete the data 20, 20, 20, 20, 20**

Key operation: 20 [SHIFT] [;] 5 [SHIFT] [CL]

• Performing single-variable statistical calculations

Once you input the data, simple display the Single-Variable Statistics Menu and select the type of result you want to view.

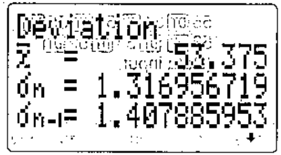
[FUNCTION] [7] (STAT)



"1. $\bar{x}$" .... Mean of x data
"2. $x\sigma_n$" .... Population standard deviation of x data
"3. $x\sigma_{n-1}$" .... Sample standard deviation of x data
"4. Σx^2" .... Sum of squares of x data
"5. Σx" .... Sum of x data
"6. n" .... Number of data items
"7. t (" .... Use when calculating a t-test value.

You can also display a comprehensive list of results by performing the following key operation.

[FUNCTION] [8] (RESULTS)



[▼] ↓ ↑ [▲]

```
Sum
Σx²=        22805
Σx =          427
n  =            8
```

- Statistical calculation results can be up to 12 digits long.
- The values for Σx^2, Σx, and n are automatically assigned to variables U, V, and W, respectively. Note that you should not use these three variables for assignment of other values when performing single-variable statistical calculations.
- Data mean and standard deviation are calculated using the following formulas.

• Mean

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{\Sigma x}{n}$$

• Standard Deviation

$$x\sigma_n = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}} = \sqrt{\frac{\Sigma x^2 - (\Sigma x)^2/n}{n}}$$

[Using all data from a finite population to determine the standard deviation for the population]

$$x\sigma_{n-1} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}} = \sqrt{\frac{\Sigma x^2 - (\Sigma x)^2/n}{n-1}}$$

[Using sample data from a population to determine the standard deviation for the population]

Example **Calculate statistical results for the data 55, 54, 51, 55, 53, 53, 54, 52. Also find the unbiased variance, and the deviation of each individual data item from the mean.**

[FUNCTION] [6] (DSP/CLR)
[6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
   operation 0
```

55 [DT] 54 [DT] 51 [DT] 55 [DT]
53 [DT] [DT] 54 [DT] 52 [DT]
(Data input)

```
52
X=     52
f=      1
n=      8
```


[FUNCTION] [8] (RESULTS)
(Statistical calculation result display)

```
Deviation
x̄    =       53.375
σn   =  1.316956719
σn-1 =  1.407885953
```

[▼]

```
Sum
Σx²=        22805
Σx =          427
n  =            8
```

[EXIT] [FUNCTION] [7] (STAT)
[3] ($x\sigma_{n-1}$) [x^2] [EXE]
(Unbiased variance)

```
xσn-1²
          1.982142857
```

55 [−] [FUNCTION] [7] (STAT)
[1] ($\bar{x}$) [EXE]
(Deviation from mean)

```
55-x̄
              1.625
```

54 [−] [FUNCTION] [7] (STAT)
[1] ($\bar{x}$) [EXE]

```
54-x̄
              0.625
```

⋮

7-2 Calculating a t-Test Value

The mean (sample mean) and sample standard deviation can be used to obtain a t-test value.

• What is the t-test?

With the t-test, sample mean x and sample standard deviation $x\sigma_{n-1}$ are used to judge, within a specific level of significance range, whether or not population mean μ is some hypothetical value.

- The t-test value is calculated using the following expression.

$$t = \frac{(\bar{x} - \mu)}{\dfrac{x\sigma_{n-1}}{\sqrt{n}}}$$

$\bar{x}$: mean of x data
$x\sigma_{n-1}$: sample standard deviation of x data
n : number of data items
μ : hypothetical population standard deviation

Example **To determine whether the population standard deviation for sample data 55, 54, 51, 55, 53, 53, 54, 52 is 53. Perform a t-test with a level of significance of 5%.**

[FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
                0
```

55 [DT] 54 [DT] 51 [DT] 55 [DT]
53 [DT] [DT] 54 [DT] 52 [DT]
(Data input)

```
52
x=             52
f=              1
n=              8
     SD
```

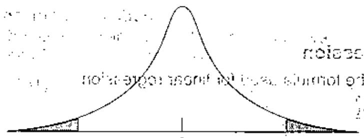
[FUNCTION] [7] (STAT)
[7] (t ( ) 53 [)] [EXE]
(Extraction of t-test value)

```
t(53)
     0.7533708035
```

The above operation produces a t-test value of $t(53) = 0.7533708035$. According to the t-distribution table below, a level of significance of 5% and a degree of freedom of 7 ($n - 1 = 8 - 1 = 7$) produces a two-sided t-test value of approximately 2.365. Since the calculated t-test value is lower than the table value, the hypothesis that population mean μ equals 53 is accepted.

• t-distribution table

The values in the top row of the table show the probability (two-sided probability) that the absolute value of t is greater than the table values for a given degree of freedom.



P (Probability) / Degree of Freedom	0.2	0.1	0.05	0.01
1	3.078	6.314	12.706	63.657
2	1.886	2.920	4.303	9.925
3	1.638	2.353	3.182	5.841
4	1.533	2.132	2.776	4.604
5	1.476	2.015	2.571	4.032
6	1.440	1.943	2.447	3.707
7	1.415	1.895	2.365	3.499
8	1.397	1.860	2.306	3.355
9	1.383	1.833	2.262	3.250
10	1.372	1.812	2.228	3.169
15	1.341	1.753	2.131	2.947
20	1.325	1.725	2.086	2.845
25	1.316	1.708	2.060	2.787
30	1.310	1.697	2.042	2.750
35	1.306	1.690	2.030	2.724
40	1.303	1.684	2.021	2.704
45	1.301	1.679	2.014	2.690
50	1.299	1.676	2.009	2.678
60	1.296	1.671	2.000	2.660
80	1.292	1.664	1.990	2.639
120	1.289	1.658	1.980	2.617
240	1.285	1.651	1.970	2.596
∞	1.282	1.645	1.960	2.576

7-3 Paired Variable Statistical Calculations

The LR Mode provides you with all the tools you need for performing regression calculations.

■ Linear Regression

The following is the formula used for linear regression.

$$y = A + Bx$$

• To input linear regression data

1. First perform the following operation to clear statistical memory, which consists of variables P, Q, R, U, V, and W.

   [FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]

• Always make sure you clear statistical memory before performing any statistical calculation.

2. Input the data using the following syntax:

   <x data value> [,] <y data value> [DT]

Example **To input the data 10/20, 20/30**
Key operation: 10 [,] 20 [DT] 20 [,] 30 [DT]

• You can input repeated data by simply pressing [DT] again.

Example **To input the data 10/20, 10/20**
Key operation: 10 [,] 20 [DT] [DT]

• You can input multiple identical data by specifying the number of repeats.

Example **To input the data 20/30, 20/30, 20/30, 20/30, 20/30**
Key operation: 20 [,] 30 [SHIFT] [;] 5 [DT]

Number of repeats
Data

• Deleting liner regression data

The method you use to delete data depends on whether or not you have already stored the data by pressing the [DT] key.

To delete data that is not stored

If you have input data on the keyboard but have not yet stored it by pressing [DT], simply press [AC].

To delete data that is already stored

Use the following syntax to delete data that you have already stored by pressing [DT]:

<x data value> [,] <y data value> [SHIFT] [CL]

Example **To delete the data 10/20 and 20/30**
Key operation: 10 [,] 20 [SHIFT] [CL] 20 [,] 30 [SHIFT] [CL]

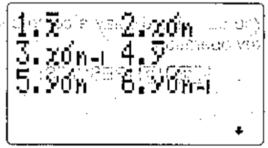
• You can delete multiple identical data by specifying the number of data items.

Example **To delete the data 20/30, 20/30, 20/30, 20/30, 20/30**
Key operation: 20 [,] 30 [SHIFT] [;] 5 [SHIFT] [CL]

• Performing regression calculations

1. Once you input the data, simple display the first Paired-Variable Statistics Menu and select the type of result you want to view.

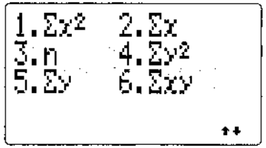
   [FUNCTION] [7] (STAT)



"1. $\bar{x}$" .... Mean of x data
"2. $x\sigma_n$" .... Population standard deviation of x data
"3. $x\sigma_{n-1}$" .... Sample standard deviation of x data
"4. $\bar{y}$" .... Mean of y data
"5. $y\sigma_n$" .... Population standard deviation of y data
"6. $y\sigma_{n-1}$" .... Sample standard deviation of y data

2. Press [▼] to advance to the second Paired-Variable Statistics Menu.

   [▼]

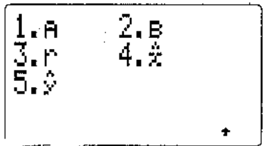


"1. Σx^2" .... Sum of squares of x data
"2. Σx" .... Sum of x data
"3. n" .... Number of data items
"4. Σy^2" .... Sum of squares of y data
"5. Σy" .... Sum of y data
"6. Σxy" .... Sum of x data and y data

• You can return to the first Paired-Variable Statistics Menu by pressing [▲].

3. Press [▼] to advance to the third Paired-Variable Statistics Menu.

   [▼]



"1. A" .... Regression formula constant term A
"2. B" .... Regression formula regression coefficient B
"3. r" .... Correlation coefficient r
"4. $\hat{x}$" .... Estimated value of x
"5. $\hat{y}$" .... Estimated value of y

- You can return to the second Paired-Variable Statistics Menu by pressing [▲].
- You can also display a comprehensive list of results by performing the following key operation.

[FUNCTION] [8] (RESULTS)

```
Regression
A  =        997.4
B  =         0.56
r  = 0.982607368
```

[▼] ↓ ↑ [▲]

```
Deviation x
x̄  =           20
σn  = 7.071067811
σn-1= 7.90569415
```

[▼] ↓ ↑ [▲]

```
Deviation y
ȳ  =       1008.6
σn  = 4.029888335
σn-1= 4.50555213
```

[▼] ↓ ↑ [▲]

```
Sum x
Σx²=        2250
Σx =         100
n  =           5
```

[▼] ↓ ↑ [▲]


```
Sum y
Σy²=     5086451
Σy =        5043
Σxy=      101000
```

- Statistical calculation results can be displayed up to 10 digits long. (Results are displayed using a 6-digit mantissa and a 2-digit exponent when exponential notation is in use.)
- The values for Σx^2, Σx, n, Σy^2, Σy, and Σxy are automatically assigned to variables U, V, W, P, Q, and R respectively. Note that you should not use these variables for assignment of other values when performing paired-variable statistical calculations.
- Constant term A, regression coefficient B, correlation coefficient r, and estimated values for x and y are calculated using the following formulas.

$$A = \frac{\Sigma y - B\cdot\Sigma x}{n} \qquad B = \frac{n\cdot\Sigma xy - \Sigma x\cdot\Sigma y}{n\cdot\Sigma x^2 - (\Sigma x)^2}$$

$$r = \frac{n\cdot\Sigma xy - \Sigma x\cdot\Sigma y}{\sqrt{\{n\cdot\Sigma x^2 - (\Sigma x)^2\}\{n\cdot\Sigma y^2 - (\Sigma y)^2\}}}$$

$$\hat{y} = A + Bx \qquad \hat{x} = \frac{y - A}{B}$$

Example **Perform linear regression to determine the regression formula terms and correlation coefficient for the following data. Next, use the regression formula to estimate atmospheric pressure at 18°C and temperature at 1000hpa.**

Temperature (°C)	Atmospheric pressure (hpa)
10	1003
15	1005
20	1010
25	1011
30	1014

[FUNCTION] [6] (DSP/CLR)
[6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
               0
```

10 [,] 1003 [DT] 15 [,] 1005 [DT]
20 [,] 1010 [DT] 25 [,] 1011 [DT]
30 [,] 1014 [DT]
(Data input)

```
30,1014
x=            30
y=          1014
n=             5
LR
```

[FUNCTION] [8] (RESULTS)
(Statistical calculation result display)

```
Regression
A  =         997.4
B  =          0.56
r  = 0.982607368
```

[EXIT] 18 [FUNCTION] [7] (STAT) [▼] [▼]
[5] ($\hat{y}$) [EXE]
(Atmospheric pressure at 18°C)

```
18ŷ
          1007.48
```

1000 [FUNCTION] [7] (STAT) [▼] [▼]
[4] ($\hat{x}$) [EXE]
(Temperature at 1000hpa)

```
1000x̂
      4.642857143
```

■ Other Regression Calculations

The linear regression formula (y = A + Bx) can also be adjusted to allow calculation of logarithmic, exponential, and power regression.

Important

- The following sections on logarithmic, exponential, and power regression assume that you are already familiar with the basic data input and editing procedures for linear regression. If you have not already done so, go to page 96 and read the section titled "Linear Regression".

■ Logarithmic Regression

The following is the formula used for logarithmic regression.

y = A + B·ln$x$

• To input logarithmic regression data

1. First perform the following operation to clear statistical memory, which consists of variables P, Q, R, U, V, and W.

   [FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]

- Always make sure you clear statistical memory before performing any statistical calculation.

2. Input the data using the following syntax:

   [ln] <x data value> [,] <y data value> [DT]

- You can input repeated data and multiple identical data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting x data values.

• Deleting logarithmic regression data

You can delete data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting x data values.

• Performing logarithmic regression calculations

Logarithmic regression calculations are performed in accordance with the following formulas.

- **Estimated value of $x = e^{\hat{y}}$**
- **Estimated value of y = ln$x\hat{y}$**

If we substitute x for ln$x$ in the logarithmic regression formula y = A + B·ln$x$, it becomes identical to the linear regression formula y = A + Bx. This means that constant term A, regression coefficient B, correlation coefficient r, and estimated values for x and y can be calculated using the same formulas as used for linear regression calculations. Note, however, that calculation results differ as shown in the following table.

Linear Regression	Logarithmic Regression
Σx	$\Sigma \ln x$
Σx^2	$\Sigma(\ln x)^2$
Σxy	$\Sigma \ln x \cdot y$

Example **Perform logarithmic regression to determine the regression formula terms and correlation coefficient for the following data. Next, use the regression formula to estimate values for x and y when xi = 80 and yi = 73.**

xi	yi
29	1.6
50	23.5
74	38.0
103	46.4
118	48.9

[FUNCTION] [6] (DSP/CLR)
[6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
                 0
```

[ln] 29 [,] 1.6 [DT] [ln] 50 [,] 23.5 [DT]
[ln] 74 [,] 38.0 [DT] [ln] 103 [,] 46.4 [DT]
[ln] 118 [,] 48.9 [DT]
(Data input)

```
ln 118,48.9
X=  4.770684624
Y=         48.9
n=            5
```

[FUNCTION] [8] (RESULTS)
(Statistical calculation result display)

```
Regression
A =-111.1283976
B =  34.0201475
r = 0.994013946
```

[EXIT] [ln] 80
[FUNCTION] [7] (STAT) [▼] [▼] [5] ($\hat{y}$) [EXE]
(Estimated value of $\hat{y}$ when xi = 80)

```
ln 80ŷ
     37.94879482
```

73 [FUNCTION] [7] (STAT) [▼] [▼] [4] ($\hat{x}$) [EXE]
[SHIFT] [e^x] [SHIFT] [Ans] [EXE]
(Estimated value of $\hat{x}$ when yi = 73)

```
73x̂
     5.412333901
eAns
     224.1541313
```

■ Exponential Regression

The following is the formula used for exponential regression.

$y = A \cdot e^{Bx}$ (ln$y$ = lnA + Bx)

• To input exponential regression data

1. First perform the following operation to clear statistical memory, which consists of variables P, Q, R, U, V, and W.

   [FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]

- Always make sure you clear statistical memory before performing any statistical calculation.

2. Input the data using the following syntax:

   <x data value> [,] [ln] <y data value> [DT]

- You can input repeated data and multiple identical data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting y data values.

• Deleting exponential regression data

You can delete data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting y data values.

• Performing exponential regression calculations

Exponential regression calculations are performed in accordance with the following formulas.

- **Constant term A = e^A**
- **Estimated value of x = ln$y\hat{x}$**
- **Estimated value of y = $e^{x\hat{y}}$**

If we substitute y for ln$y$ and a for lnA in the exponential regression formula $y = A \cdot e^{Bx}$ (ln$y$ = lnA + Bx), it becomes identical to the linear regression formula y = A + Bx. This means that constant term A, regression coefficient B, correlation coefficient r, and estimated values for x and y can be calculated using the same formulas as used for linear regression calculations. Note, however, that calculation results differ as shown in the following table.

Linear Regression	Exponential Regression
Σy	Σln$y$
Σy^2	Σ(ln$y$)^2
Σxy	$\Sigma x \cdot$ln$y$

Example **Perform exponential regression to determine the regression formula terms and correlation coefficient for the following data. Next, use the regression formula to estimate values for x and y when xi = 16 and yi = 20.**

xi	yi
6.9	21.4
12.9	15.7
19.8	12.1
26.7	8.5
35.1	5.2

[FUNCTION] [6] (DSP/CLR)
[6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
               0
```

6.9 [,] [ln] 21.4 [DT]
12.9 [,] [ln] 15.7 [DT]
19.8 [,] [ln] 12.1 [DT]
26.7 [,] [ln] 8.5 [DT]
35.1 [,] [ln] 5.2 [DT]
(Data input)

```
35.1,ln 5.2
X=          35.1
Y=    1.64865862
n=             5
```

[FUNCTION] [8] (RESULTS)
(Statistical calculation result display)

```
Regression
A  = 3.417647579
B  =-0.049203708
r  =-0.997247351
```

[EXIT] 16 [FUNCTION] [7] (STAT) [▼] [▼]
[5] ($\hat{y}$) [EXE]
[SHIFT] [e^x] [SHIFT] [Ans] [EXE]
(Estimated value of $\hat{y}$ when xi = 16)

```
16ŷ
     2.630388247
eAns
     13.87915739
```

[ln] 20 [FUNCTION] [7] (STAT) [▼] [▼]
[4] ($\hat{x}$) [EXE]
(Estimated value of $\hat{x}$ when yi = 20)

```
ln 20x̂
     8.574868047
```

■ Power Regression

The following is the formula used for power regression.

$y = A \cdot x^B$ (ln$y$ = lnA + Bln$x$)

• To input power regression data

1. First perform the following operation to clear statistical memory, which consists of variables P, Q, R, U, V, and W.

   [FUNCTION] [6] (DSP/CLR) [6] (Scl) [EXE]

• Always make sure you clear statistical memory before performing any statistical calculation.

2. Input the data using the following syntax:

   [ln] <x data value> [,] [ln] <y data value> [DT]

• You can input repeated data and multiple identical data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting x data values and y data values.

• Deleting power regression data

You can delete data using the same basic procedures as described for linear regression on page 96. Just remember to press [ln] before inputting x data values and y data values.

• Performing power regression calculations

Power regression calculations are performed in accordance with the following formulas.

• **Constant term A = e^A**
• **Estimated value of $x = e^{\ln \hat{y}}$**
• **Estimated value of $y = e^{\ln x \hat{y}}$**

If we substitute y for ln$y$, a for lnA, and x for ln$x$ in the power regression formula $y = A \cdot x^B$ (ln$y$ = lnA + Bln$x$), it becomes identical to the linear regression formula $y = A + Bx$. This means that constant term A, regression coefficient B, correlation coefficient r, and estimated values for x and y can be calculated using the same formulas as used for linear regression calculations. Note, however, that calculation results differ as shown in the following table.

Linear Regression	Power Regression
Σx	$\Sigma \ln x$
Σx^2	$\Sigma(\ln x)^2$
Σy	$\Sigma \ln y$
Σy^2	$\Sigma(\ln y)^2$
Σxy	$\Sigma \ln x \cdot \ln y$

Example **Perform power regression to determine the regression formula terms and correlation coefficient for the following data. Next, use the regression formula to estimate values for x and y when xi = 40 and yi = 1000.**

xi	yi
28	2410
30	3033
33	3895
35	4491
38	5717

[FUNCTION] [6] (DSP/CLR)
[6] (Scl) [EXE]
(Clears statistical memory.)

```
Scl
               0
```

[ln] 28 [,] [ln] 2410 [DT]
[ln] 30 [,] [ln] 3033 [DT]
[ln] 33 [,] [ln] 3895 [DT]
[ln] 35 [,] [ln] 4491 [DT]
[ln] 38 [,] [ln] 5717 [DT]
(Data input)

```
ln 38,ln 5717
X=   3.637586159
Y=   8.651199471
n=             5
```

[FUNCTION] [8] (RESULTS)
(Statistical calculation result display)

```
Regression
A  =-1.432124422
B  = 2.771866157
r  = 0.998906255
```

- The value for constant term A in the above display is calculated for e^A. Use the following key operation to calculate A for $A \cdot x^B$.

[EXIT] [SHIFT] [e^x] [FUNCTION] [7] (STAT) [▼] [▼]
[1] (A) [EXE]

```
eA
      0.2388010685
```

[ln] 40 [FUNCTION] [7] (STAT) [▼] [▼]
[5] ($\hat{y}$) [EXE]
[SHIFT] [e^x] [SHIFT] [Ans] [EXE]
(Estimated value of $\hat{y}$ when xi = 40)

```
ln 40ŷ
       8.792955696
eAns
       6587.674589
```

[ln] 1000 [FUNCTION] [7] (STAT) [▼] [▼]
[4] ($\hat{x}$) [EXE]
[SHIFT] [e^x] [SHIFT] [Ans] [EXE]
(Estimated value of $\hat{x}$ when yi = 1000)

```
ln 1000x̂
       3.008759885
eAns
       20.26225681
```


Chapter 8

Formula Storage

Chapter 8 Formula Storage

You can store one formula in memory for instant recall when you need it. Then you can recall the formula at any time, plug in values, and perform calculations quickly and easily.
The following are the key operations you can use for formula storage, recall, and execution.

[SHIFT] [IN] ...... Stores the formula on the display in formula memory.
[OUT] .... Recalls formula memory contents.
[CALC] .... Starts a calculation operation.

8-1 Using Formula Memory

The best way to explain how to use formula memory is with the aid of a couple of actual examples.

Example 1 **To store the following formula in formula memory, and then use it to perform a calculation.**
$Y = AX^2 + 6X - 9$

1. Input the formula.

[AC] [SHIFT] [ALPHA] [Y] [=] [A] [X] [ALPHA] [x^2] [+] [6] [ALPHA] [X] [−] [9]

```
Y=AX²+6X-9_
```

2. Store it into formula memory.

[SHIFT] [IN]

```
_
```

3. Start the calculation operation.

[CALC]

```
Y=AX²+6X-9
■A=          0
 X=          0
```

Cursor

Values currently assigned to variables

- The cursor indicates the variable currently selected for input. Use [▲] and [▼] to move the cursor. The cursor also automatically moves to the next variable whenever you press [EXE].
- If the cursor is located next to the last variable, pressing [EXE] executes the calculation.

[2] [EXE]
(Value of variable A)

[5] [EXE]
(Value of variable X)

```
Y=AX²+6X-9
 A=          2
■X=          5
```


[CALC]
(Calculation of result)

```
Y=AX²+6X-9
            71

REPEAT:[EXE]
```

[EXE]
(Restarts operation from the beginning.)

```
Y=AX²+6X-9
 A=          2
■X=          5
```

- If the calculation is taking too much time (such as when you are using a multistatement to perform a series of calculations), you can interrupt it by pressing [AC]. This causes the message "**Calculation Stopped**" appears on the display. Press [AC] again to clear the message.

Example 2 **To recall the formula $Y = AX^2 + 6X - 9$ and change it to $Y = AX^2 + 3X - 9$.**

1. Recall the formula you want to edit.

[OUT]

```
Y=AX²+6X-9_
```

2. Move the cursor to the location where you want to make your change.

[◄] [◄] [◄] [◄]

```
Y=AX²+6X-9
```

3. Make the changes you want, and then store the formula.

[3]

```
Y=AX²+3X-9
```

[SHIFT] [IN]

- To delete a formula from memory, recall it and press [AC] [SHIFT] [IN].

Important

- You can store only one formula in formula memory. Note that a multistatement (page 28) made up of multiple formulas is treated as a single formula.
- Storing a formula into formula memory automatically deletes anything that was previously in the formula memory.
- Formula memory can hold up to 127 bytes of data.
- The current calculation mode (COMP, SD, LR, BASE-N) of the calculator is also stored as part of the formula data.
- When you use an array variable inside of an formula, the value currently assigned to the variable is used when you perform a calculation using the formula.
- A "**BASE-N Mode ERROR!**" occurs whenever you are in the BASE-N Mode and try to recall a formula that was stored while in another mode, or whenever you are not in the BASE-N Mode and try to recall a formula that was stored while in the BASE-N mode.
- Formula memory contents are retained even while the calculator is turned off.

8-2 Comment Text

You can append comment text to variables used in formulas you store in formula memory. Simply input the text you want to use inside of double quotation marks after the variable name. Once you append comment text to a variable, the text appears on the display whenever you execute the calculation.

- Comment text can be up to 15 characters long.

Example **To store the following formula in formula memory.**
S"AREA" = 3.14 × R^2

1. Input the formula.

[AC] [SHIFT] [ALPHA] [S] ["] [A] [R] [E] [A] ["] [=] [ALPHA] [3] [•] [1] [4] [×] [ALPHA] [R] [x^2]

S"AREA"=3.14×R²_

[SHIFT] [IN]

2. Perform a calculation.

[CALC]

S"AREA"=3.14×R²
■R= 0

[3] [EXE]
(Value assigned to variable R)
[CALC]

AREA=
28.26
REPEAT:[EXE]

⋮

8-3 Table Function

The Table Function lets you assign a range of values to a single variable in a formula stored in formula memory. You can then calculate the results for the entire range of values.

Example **To assign values starting from 0 and incrementing by 2 to X in the following formula (A = 2).**

Y = AX^2 + 6X − 9

1. Input the formula.

[AC] [SHIFT] [ALPHA] [Y] [=] [A] [X] [ALPHA] [x^2] [+] [6] [ALPHA] [X] [−] [9]

Y=AX²+6X−9_

[SHIFT] [IN]

2. Press [CALC] and then input the value for A:

[CALC] [2] [EXE]

Y=AX²+6X−9
A= 2
■X= 0

3. When the cursor is next to the variable to which you want to assign a range of values (X in our example), press [SHIFT] [TBL]. This causes a table range start value specification screen to appear.

[SHIFT] [TBL]

TABLE Range
X
Start?
1

4. Input the starting value you want to use for X and press [EXE].

[0]

Start?
0_

[EXE]

TABLE Range
X
Pitch?
1

5. In the next screen that appears, input a value for the pitch, which is how much the value of X will change with each execution. Next, press [EXE] to perform the first execution using the starting value.

[2]

Pitch?
2_

[EXE]

Y=AX²+6X−9
-9
X=
0

6. Continue pressing [EXE] as many times as you want.

[EXE]

```
Y=AX²+6X-9
           11
X=
            2
```

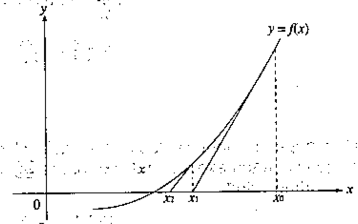
⋮

- A positive value for the pitch will cause the value of the variable to increase. A negative value will cause the value to decrease.
- A message indicating an Ma ERROR appears on the display whenever the calculation result is out of range or when the calculation result is an imaginary number.

8-4 Solve Function

The Solve Function uses Newton's method to solve for any of the variables of a formula stored in formula memory.

Newton's Method



This method is based on the assumption that $f(x)$ can be approximated by a linear expression within a very narrow range. It is used to approximate the roots of an equation by the methods of the calculus.
First, a starting value (predicted value) x_0 is given. Using this starting value as a base, approximate value x_1 is obtained, and then the left side and right side calculation results are compared. Next, approximate value x_1 is used as the initial value to calculate the next approximate value x_2. This procedure is repeated until the difference between the left side and right side calculated values is less than some minute value.

Example To calculate the value of X when Y = 0 and A = 2 in the following formula:

Y = AX² + 6X − 9

1. Input the formula.

[AC] [SHIFT] [ALPHA] [Y] [=] [A] [X] [ALPHA] [x²] [+] [6] [ALPHA] [X] [−] [9]

[SHIFT] [IN]

```
Y=AX²+6X-9_
```

2. Press [SOLVE].

[SOLVE]

```
Y=AX²+6X-9
■Y=         0
 A=         0
 X=         0
```

3. Input the values for variables Y and A.

[0] [EXE] (Value of Y)
[2] [EXE] (Value of A)

```
Y=AX²+6X-9
 Y=         0
 A=         2
■X=         0
```

4. Move the cursor next to the variable you want to solve for, and press [SOLVE].

[SOLVE]

```
Y=AX²+6X-9
X=  1.098076211
Lft=          0
Rgt=     -8.E-14
```

- Lft shows the calculated left side value while Rgt shows the calculated right side value. The closer the difference between these values is zero, the greater the precision of the calculation result.

[SOLVE]
(Re-executes the calculation)

```
Y=AX²+6X-9
■Y=          0
 A=          2
 X=  1.098076211
```

⋮

- Since the Solve Function uses Newton's Method, certain initial values (assumed values) can make it impossible to obtain solutions. In this case, try inputting another value that you assume to be near the solution and perform the calculation again.
- The Solve Function may be unable to obtain a solution, even though a solution exists.
- Due to certain idiosyncrasies of Newton's method, solutions for the following types of functions tend to be difficult to calculate.
  Periodic functions (i.e. $y = \sin x$)
  Functions whose graph produce sharp slopes (i.e. $y = e^x$, $y = 1/x$)
  Discontinuous functions (i.e. $y = \sqrt{x}$)
- Solutions obtained using the Solve Function may include errors.
- To interrupt an ongoing Solve Function operation, press the [EXIT] key.

Important

- The message **"Try again: [EXE]"** appears along with any value that is an intermediate result and not the final solution. When this happens, press [EXE] to continue the calculation using the value on the display.
- The message shown nearby appears whenever the calculator is unable to calculate a result. In this case, press [AC] or [EXIT] to clear the message.

```
**Can't solve**
Adjust
  initial value.
Then try again.
```

8-5 Storing Formulas in the Program Area

You can copy formula memory contents into the calculator's program area under a file name. This let's you store multiple formulas for recall when you need them.

- **To store a formula in the program area**

Example **To store the following formula (which is already stored in formula memory) in the program area under the name "QUADRATIC".**

$Y = AX^2 + 6X - 9$

1. Press [MODE] [5] (PROG) to display the Program Menu.

   [MODE] [5] (PROG)

```
Program menu
1.NEW   2.RUN
3.EDIT  4.DELETE
 4300 Bytes Free
```


2. Press [1] (NEW) and then input the file name you want to use.

   [1] (NEW)
   [Q] [U] [A] [D] [R] [A] [T] [I] [C]

```
Filename?
[QUADRATIC_  ]
```

3. Press [EXE] to register the file name.

   [EXE]

```
PGM:QUADRATIC
1.COMP  2.BASE-N
3.SD    4.LR
5.Save formula
```

4. Press [5], and a confirmation message appears.

   [5]

```
PGM:QUADRATIC
Save formula?
 YES:[EXE]
 NO :[EXIT]
```

5. Press [EXE] to store the formula and return to the Program Menu.

- To abort the save operation without saving anything, press [EXIT] instead of [EXE].

- **To recall a formula from the program area**

Example **To recall to formula memory the formula stored in the program area under the file name "QUADRATIC".**

1. While in the COMP, BASE-N, SD, or LR Mode, press [FILE], or input [2] (RUN) while the Program Menu is on the display.

   [2] (RUN)

```
Program [RUN]
■OCTAHEDRON  :CO
 TRIANGLE    :CO
 QUADRATIC   :CO
               ↓
```

Cursor

The above display shows multiple files are stored in the program area.

2. Use ▼ and ▲ to move the cursor next to the file name of the formula you want to recall.

3. Press SHIFT IN, and a confirmation message appears on the display.

SHIFT IN

```
PGM:QUADRATIC
Load formula?
 YES:[EXE]
 NO :[EXIT]
```

4. Press EXE to recall the formula and store it in formula memory.

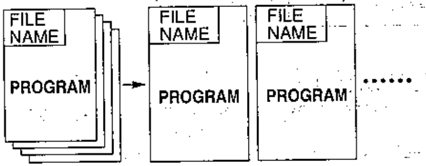
- Remember that recalling a formula and storing it in formula memory will delete anything that is already stored in the formula memory.
- To abort the recall operation without recalling anything, press EXIT instead of EXE.

Chapter 9

Programming

Chapter 9 Programming

You can store often-used and complex formulas under file names in the calculator's program area. Multiple calculations can be linked to form multistatements (page 28) that perform a series of calculations. Up to 4,500 bytes of data can be stored in the program area.



9-1 Before Using the Program Area

You have to use the following procedure to enter the **PROG Mode** from the Main Menu in order to use the program area.

[MODE] [5] (PROG)

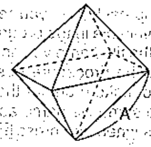
```
Program menu
1.NEW    2.RUN
3.EDIT   4.DELETE
 4300 Bytes Free
```

"1. NEW" .... Creates a new program.
"2. RUN" .... Executes a stored program (page 121).
"3. EDIT" .... For editing of a stored program (page 125).
"4. DELETE" .... Deletes a stored program (page 130).

- If there is nothing stored in the program area, the message **"No file"** appears when you enter the PROG Mode.

9-2 Storing a Program

Example **To store a program that calculates the surface area and volume of three regular octahedrons, each with one side 7cm, 10cm, and 15cm long. Store the program under the file name "OCTAHEDRON".**



The following are the formulas for calculation of the surface area (S) and volume (V) of a regular octahedron when the length of one side (A) is known.

$$S = 2\sqrt{3}A^2, \quad V = \frac{\sqrt{2}}{3}A^3$$

To store a program, you register the program name, specify the mode to use for program execution, and the input the program itself.

■ To Register a Program Name

1. While the Program Menu is on the display, press [1] (NEW) to select input of a new program.

[1] (NEW)

```
Filename?
[_          ]
```

2. Input the file name you want to use.

[O] [C] [T] [A] [H] [E] [D] [R] [O] [N]

```
Filename?
[OCTAHEDRON_ ]
```

- The cursor indicates upper-case alpha character input.
- You can input up to 12 characters for a file name. Valid characters are alpha characters A through Z, space, numbers 0 through 9, decimal point, open bracket, closed bracket, and arithmetic operator symbols (+, −, ×, ÷).
- You can delete specific characters by moving the cursor to the character to be deleted and pressing [DEL].

3. Press [EXE] to register the file name and change to the menu of program execution modes.

[EXE]

File name —

```
PGM:OCTAHEDRON
1.COMP   2.BASE-N
3.SD     4.LR
5.Save formula
```

- 17 bytes of memory are used up each time you register a file name. If there are less than 17 bytes of memory available in the program area when you press [EXE] in step 3 of the above procedure, the file name you specified will not be registered. In this case, you will have to use the procedure described on page 130 to delete programs you no longer need to make room for the new input.
- If there is already a file with the same file name as the one you specify, the message **"Already exists"** appears when you press [EXE] in step 3 of the above procedure.
- Nothing is registered if you press [EXE] without inputting a file name.
- To abort the above procedure any time before you press [EXE] in step 3, press [EXIT] to return to the Program Menu.

■ To Specify the Program Execution Mode

While the Program Menu is on the display, input the value from 1 to 4 that specifies the mode the calculator should enter when executing the program you are inputting.

```
PGM:OCTAHEDRON
1.COMP   2.BASE-N
3.SD     4.LR
5.Save formula
```

"1. COMP" .... COMP Mode
"2. BASE-N" .... BASE-N Mode
"3. SD" .... SD Mode
"4. LR" .... LR Mode
"5. Save formula" .. Saves formula memory contents under the specified file name.

Here we will press [1] (COMP) to specify the COMP Mode and move on to actual input of the program's contents.

[1] (COMP)

■ Inputting Program Contents

Program contents are input using the same basic procedures that you use for manual calculations. The following shows how you would input the required formulas as manual calculations and as a program.

- Manual Calculation
  - Surface Area S [2] [×] [√] [3] [×] <Value of A> [x²] [EXE]
  - Volume V [√] [2] [÷] [3] [×] <Value of A> [^] [3] [EXE]
- Program
  - Surface Area S [2] [×] [√] [3] [×] [ALPHA] [A] [x²] [EXE]
  - Volume V [√] [2] [÷] [3] [×] [ALPHA] [A] [^] [3] [EXE]

If you just input the two formulas, the calculator will execute them without stopping for you to see the results. For this reason you will have to use the output command to tell the calculator to stop and show you calculated results along the way. The following table describes the output command and its close cousin, the multistatement command.

◢	The display result command "◢" causes program execution to stop while the calculation result up to the display result command or a text message is displayed. To resume program execution, press [EXE]. The final result of the program execution is displayed regardless of whether or not this command is included at the end.
:	This is the multistatement command, which is used to connect two calculation expressions or commands. With the multistatement command, execution of the connected statements is performed without stopping.

- **To input program contents**

[2] [×] [√] [3] [×] [ALPHA] [A] [x²] [SHIFT] [◢]

```
2×√3×A²◢
_
```

[√] [2] [÷] [3] [×] [ALPHA] [A] [^] [3]

```
2×√3×A²◢
√2÷3×A^3_
```

[EXIT] [EXIT]

```
Program menu
1.NEW    2.RUN
3.EDIT   4.DELETE
 4467 Bytes Free
```

■ Running a Program

You can use any one of three different methods to run a program.

- From the Program Menu
- Pressing the [FILE] key
- Pressing [SHIFT] [Prog]

- **To run a program from the Program Menu**

1. While the Program Menu is on the display, press [2] (RUN).

```
Program [RUN]
■OCTAHEDRON  :CO
```

Cursor — Program execution mode

• Program execution modes are indicated as CO (COMP) and BN (BASE-N).

2. Use [▲] and [▼] to move the cursor next to the name of the program you want to run.

Length of One Side (A)	Surface Area (S)	Volume (V)
7cm	169.7409791cm²	161.6917506cm³
10cm	346.4101615cm²	471.4045208cm³
15cm	779.4228634cm²	1590.990258cm³

3. Press [EXE] to run the program.

```
A?
                 0
```

[7] [EXE]
(Value of A)

```
                 0
7
2×√3×A²
       169.7409791
              Disp
```

• [Disp] indicates that program execution is paused for display of an intermediate result.

[EXE]

```
2×√3×A²
       169.7409791
√2×3×A^3
       161.6917506
```

[EXE]

```
√2÷3×A^3
       161.6917506
A?
                 7
```

[1] [0] [EXE]

```
                 7
10
2×√3×A²
       346.4101615
              Disp
```

[EXE]

```
2×√3×A²
       346.4101615
√2÷3×A^3
       471.4045208
```

• Pressing [EXE] again while the final calculation result is on the display re-runs the program from the beginning.

• To run a program by pressing the [FILE] key

1. While in the COMP, BASE-N, SD, or LR Mode, press [FILE].
2. Use [▲] and [▼] to move the cursor next to the name of the program you want to run.
3. Press [EXE] to run the program.

• To run a program by pressing [SHIFT] [Prog]

1. While in the COMP, BASE-N, SD, or LR Mode, press [SHIFT] [Prog].
2. Input the file name of the program in using the following syntax.
   [ALPHA] [”] <file name> [ALPHA] [”]
3. Press [EXE] to run the program.

9-3 Error Messages

Sometimes a program you enter causes an error message to appear when you execute it. This means that there is an error that needs to be corrected. The following shows a typical error message display.

```
   Ma ERROR   ← Error type
in TRIANGLE   ← File name of program where error occurred
```

All of the possible error messages are listed in the Error Message Table on page 150. When you get an error message, look it up in the Error Message Table and take actions to correct it.

9-4 Counting the Number of Bytes

The memory of this unit can hold up to 4,500 bytes of data. Generally, one function in a program takes up one byte. Some functions, however, require two bytes each.

- **1-byte functions**

sin, cos, tan, log, (,), A, B, 1, 2, etc.

- **2-byte functions**

Lbl 1, Goto 2, etc.

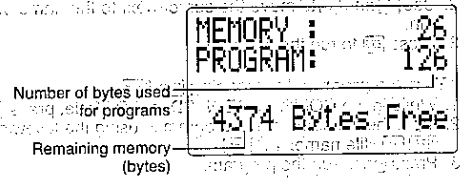
You can count the bytes in a program by pressing the [◀] and [▶] keys. Each press of these keys causes the cursor to jump one byte.

When the number of bytes remaining drops to five or below, the cursor automatically changes from an underline to "■". If you need to input more than five bytes, try to increase the amount of memory available for program storage by deleting unnecessary programs, deleting expanded memory, or by deleting unneeded function memory contents.

■ Checking the Amount of Memory Remaining

You can also display the remaining memory display by performing the following operation while the COMP, BASE-N, SD, or LR Modes are displayed.

[SHIFT] [Defm] [EXE]



9-5 Searching for a File Name

You can use either "sequential search" or "direct search" to locate a file name.

- **Sequential search**

With sequential search, you scroll through the file names on the calculator's display until you locate the one you want.

- **Direct search**

With direct search, you input the first few characters of the file name and recall the names that match.

■ To Use Sequential Search

1. While the Program Menu is on the display, press [3] (EDIT).
2. Use [▲] and [▼] to move the cursor next to the name of the program you want to run.
3. Press [EXE] to display the contents of the program.

Note

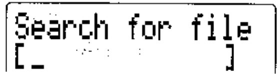
- You can also display a list of program area files by pressing [FILE] while in the COMP, BASE-N, SD, or LR Mode. In this case, however, you will not be able to view the contents of the program.

■ To Use Direct Search

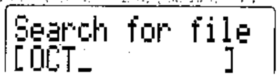
1. While the Program Menu is on the display, press [3] (EDIT).

2. Press [FUNCTION] to display the File Commands Menu.

3. Press [1] (SEARCH).



4. Input the first few characters of the file name you want to find.



5. Press [EXE] to start the search.

[EXE]

- The message "**No file**" appears on the display if there are not any file names that start with the characters you specified.

6. Use [▲] and [▼] to scroll through the list of recalled file names and move the cursor next to the file name you want.

- If there are more than four file names that start with the characters you specified, [▲] and [▼] will scroll the list of names on the display.

7. Press [EXE] to display the contents of the program.

Note

- You can also use the following direct search procedure to look for a file name while in the COMP, BASE-N, SD, or LR Mode.

1. Input the first few characters of the file name you want to find.
2. Press [FILE] to perform the search and display a list of file names whose first characters match the ones you specified.
   In this case, however, you will not be able to view the contents of the program.

9-6 Editing Program Area Data

Use the following procedures to edit file names and program contents.

■ To Edit a File Name

1. While the Program Menu is on the display, press [3] (EDIT), and move the cursor next to the file name you want to edit.

```
Program [EDIT]
 OCTAHEDRON   :C0
■TRIANGLE     :C0
```

2. Press [FUNCTION] to display the File Commands Menu.

```
File Commands
1.SEARCH
2.RENAME
```

3. Press [2] (RENAME).

```
Rename file
[TRIANGLE    ]
```


4. Change the file name to the one you want.

[DEL] [DEL] [DEL]

```
Rename file
[ANGLE       ]
```

5. Press [EXE] to store the program under its new file name.

- If there is already a file with the same file name as the one you specify, the message **"Already exists"** appears when you press [EXE] in step 5 of the above procedure. If this happens, press [◄] or [►] to display the new file name. Next, press [AC] to clear the file name and input a different one.

■ To Edit Program Contents

1. While the Program Menu is on the display, press [3] (EDIT), and move the cursor next to the file name of the program whose contents you want to edit.
2. Press [EXE] to display the contents of the program.
3. Make the changes you want to the contents of the file.
- For details on how to edit data, see Editing Calculations on page 26.
4. After you are finished making the changes you want, press [EXIT] [EXIT] to store the program.

• Helpful Cursor Commands

The following cursor commands can help you move around inside a program quicker and easier.

- [SHIFT] [▲]

This command makes the cursor jump to the beginning of the program.

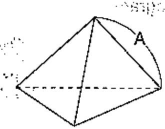
```
2×√3×A²↵
√2÷3×A^3
```

- [SHIFT] [▼]

This command makes the cursor jump to the end of the program.

```
2×√3×A²↵
√2÷3×A^3_
```

Example **To edit the program named OCTAHEDRON (which in stored on page 118) so that it calculates the surface area and volume of three regular tetrahedrons, each with one side 7cm, 10cm, and 15cm long. Also change the name of the program to "TETRAHEDRON".**



The following are the formulas for calculation of the surface area (S) and volume (V) of a regular tetrahedron when the length of one side (A) is known.

$$S = \sqrt{3}A^2, \qquad V = \frac{\sqrt{2}}{12}A^3$$

The following shows the differences between the OCTAHEDRON and the TETRAHEDRON programs.

OCTAHEDRON

Surface Area S [2] [×] [√] [3] [×] [ALPHA] [A] [x²] [SHIFT] [◢]

Volume V [√] [2] [÷] [3] [×] [ALPHA] [A] [^] [3]

TETRAHEDRON

Surface Area S [√] [3] [×] [ALPHA] [A] [x²] [SHIFT] [◢]

Volume V [√] [2] [÷] [1] [2] [×] [ALPHA] [A] [^] [3]

As you can see, we need to delete [2] [×] at the beginning of the program, and to change [3] to [1] [2].

• Procedure

1. Change the name of the program.

[MODE] [5] (PROG) [3] (EDIT)

```
Program [EDIT]
■OCTAHEDRON  :CO
```

[FUNCTION]

```
File Commands
1.SEARCH
2.RENAME
```

[2] (RENAME)
[T] [E] [T] [R] [A] [H] [E] [D] [R] [O] [N]

```
Rename file
[TETRAHEDRON_]
```

[EXE]

```
Program [EDIT]
■TETRAHEDRON :CO
```

2. Change the contents of the program.

[EXE]

```
2×√3×A²◢
√2÷3×A^3
```


[DEL] [DEL]

```
√3×A²◢
√2÷3×A^3
```

[▼] [▶] [▶] [▶]
[SHIFT] [INS] [1] [2]

```
√3×A²◢
√2÷123×A^3
```

[DEL]

```
√3×A²◢
√2÷12×A^3
```

[EXIT] [EXIT]

```
Program menu
1.NEW    2.RUN
3.EDIT   4.DELETE
 4468 Bytes Free
```

3. Now run the program.

Length of One Side (A)	Surface Area (S)	Volume (V)
7cm	84.87048957cm²	40.42293766cm³
10cm	173.2050808cm²	117.8511302cm³
15cm	389.7114317cm²	397.7475644cm³

[2] (RUN) [EXE]

```
A?
             0
```

[7] [EXE]
(Value of A)

```
             0
7
√3×A²
   84.87048957
          Disp
```

[EXE]

```
√3×A²
   84.87048957
√2÷12×A^3
   40.42293766
```

[EXE]

```
√2÷12×A^3
          40.42293766
A?
                    7
```

[1] [0] [EXE]

```
                    7
10
√3×A²
          173.2050808
```

[EXE]

```
√3×A²
          173.2050808
√2÷12×A^3
          117.8511302
```

⋮ ⋮

9-7 Deleting Programs

You can delete a specific program or all programs stored in the program area.

■ To Delete a Specific Program

Example **To delete the program named TRIANGLE.**

1. While the Program Menu is on the display, press [4] (DELETE).

   [4] (DELETE)

```
Delete Program
1.ONE PROGRAM
2.ALL PROGRAMS
```

2. Press [1] (ONE PROGRAM).

   [1] (ONE PROGRAM)

Currently selected program

```
Program [DELETE]
■OCTAHEDRON  :CO
 TRIANGLE    :CO
 OCTONARY    :BN
```


3. Move the cursor next to the file name of the program you want to delete. Press [EXE] and a confirmation message appears on the display.

   [▼] [EXE]

```
PGM:TRIANGLE
Delete?
 YES:[EXE]
 NO :[EXIT]
```

4. Press [EXE] to delete the program.

   [EXE]

```
Program [DELETE]
 OCTAHEDRON  :CO
■OCTONARY    :BN
 PRIME FACTOR:CO
```

- To abort the delete operation without deleting anything, press [EXIT] instead of [EXE].

■ To Delete All Programs

Example **To delete all the programs.**

1. While the Program Menu is on the display, press [4] (DELETE).

   [4] (DELETE)

```
Delete Program
1.ONE PROGRAM
2.ALL PROGRAMS
```

2. Press [2] (ALL PROGRAMS) and a confirmation message appears on the display.

   [2] (ALL PROGRAMS)

```
**All programs**
Delete?
 YES:[EXE]
 NO :[EXIT]
```

3. Press [EXE] to delete all the programs.

[EXE]

```
Program menu
1.NEW    2.RUN
3.EDIT   4.DELETE
*** No file ****
```

• To abort the delete operation without deleting anything, press [EXIT] instead of [EXE].

9-8 Programming Commands

A collection of powerful programming commands are also available to allow you to incorporate logical operations, conditional jumps, and other sophisticated techniques into your programs.

■ Program Command Menu

You can access most of the special programming commands from the Program Command Menu.

1. Use the following operation to display the first Program Command Menu.

[FUNCTION] [3] (PROG)

```
1.⇒      2.≠⇒
3.⊿      4.Goto
5.Lbl    6.Dsz
7.Isz
```

"1. ⇒" .... Conditional jump success code
"2. ≠⇒" .... Conditional jump failure code
"3. ⊿" .... Conditional jump end code
"4. Goto" .... Unconditional jump command
"5. Lbl" .... Label command
"6. Dsz" .... Decrement command
"7. Isz" .... Increment command

2. Press [▼] to advance to the second Program Command Menu.

[▼]

```
1.Pause  2.Fixm
3.{      4.}
```

"1. Pause" .... Pause command
"2. Fixm" .... Variable lock command
"3. { " .... Variable input command
"4. } " .... Variable input command

• You can return to the first Program Command Menu by pressing [▲].

3. Press [▼] to advance to the third Program Command Menu.

[▼]

```
1.=      2.≠
3.>      4.<
5.≥      6.≤
```

"1. = " .... Conditional jump relational operator
"2. ≠ " .... Conditional jump relational operator
"3. > " .... Conditional jump relational operator
"4. < " .... Conditional jump relational operator
"5. ≥ " .... Conditional jump relational operator
"6. ≤ " .... Conditional jump relational operator

• You can return to the second Program Command Menu by pressing [▲].

■ Variable Input Command

Any value you assign to a variable in a program is normally fixed throughout the duration of the program. This means that it is normally impossible for you to assign a new value to the same variable when you use a jump command to execute the same subroutine inside a program.
In order to overcome this, you must use the variable input command to assign a different value to a variable. The variable input command unlocks a variable and lets you assign a different value to it.
To execute the variable input command inside of a program, simply enclose the variable name inside of curly braces.

Examples **{A}: Unlocks variable A.**
{AB}, {A, B}, {A B}: Unlocks variables A and B.

• A pair of curly braces is treated as one statement.
• Array variables cannot be used as variables.

■ Variable Lock Command

The Fixm variable lock command locks all variables, which means that the values currently assigned to all variable become fixed, and cannot be changed. The Fixm variable lock command can be inserted to relock variables that have been unlocked using the variable input command.
• The Fixm command is treated as one statement.
• Fixm takes priority over the variable input command.

Jump Commands

Jump commands can be used to change the flow of program execution. They can be used to cause the same expression to be executed a number of times, or to make program execution jump to another location.
The following are the three types of jump commands that are available with this calculator.

- Unconditional jump
  This type of jump is immediately executed, without any preconditions being checked.
- Conditional jump
  This type of jump sets up a certain preconditions that are tested. The jump destination is determined depending upon whether or not these conditions are satisfied.
- Count jump
  A control value is incremented or decremented with each pass, and the jump is executed when the control value reaches zero.

• Unconditional jump

Unconditional jumps are constructed using the following two commands.
 Goto *n* (where *n* = a value from 0 to 9 or a letter from A to Z)
 Lbl *n* (where *n* is the same value or letter used in the corresponding Goto *n*)
Executing the Goto *n* command causes program execution to immediately jump to the corresponding Lbl *n*.
An unconditional jump can be used to make a program jump back to its beginning and continue to run in an endless loop. Or you can use it to make only one part of a program to repeat endlessly. The unconditional jump can also be used in combination with conditional jumps and count jumps.

Example 1 **To create a program that continually calculates $y = a + bx$, with new values input for x, a, and b each time.**

Note the use of the variable input command in the following program.

```
Lbl 1
 {A, B, X}
Y = A + B × X◢
Goto 1
```

Unconditional jump

Example 2 **To create a program that continually calculates $y = a + bx$, with new values input for x, and $a = 2$ and $b = 5$.**

The following program causes values to be assigned to variables, which are then locked. When the program is executed, a value is input for x only.

```
A = 2
B = 5
Lbl 1
{X}
Y = A + B × X◢
Goto 1
```

Unconditional jump

- A **Go ERROR** occurs whenever a Goto *n* does not have a corresponding Lbl *n*.

• Conditional jump

A conditional jump compares two variables or arithmetic expressions. Based on the results of the comparison, a decision is made to next jump to either the statement following ⇒ or the statement following ≠⇒. The following is the syntax for a conditional jump.

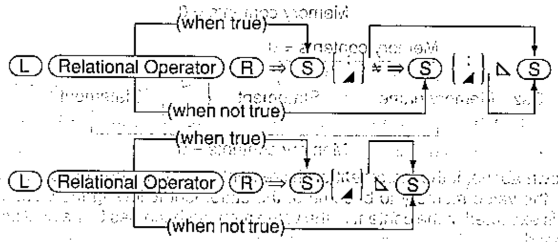
- (L) (Relational Operator) (R) ⇒ (S) { : / ◢ } ≠⇒ (S) { : / ◢ } ⊿ (S)
- (L) (Relational Operator) (R) ⇒ (S) { : / ◢ } ⊿ (S)

(L) : Left side (R) : Right side (S) : Statement

Left Side and Right Side can be variables (A through Z), constants, or variable expressions (i.e. A × 2, B − C). The following are the six relational operators that can be used in a conditional jump.

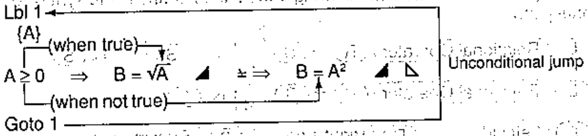
L = R .... True when L and R are equal; false when L and R are not equal
L ≠ R .... True when L and R are not equal; false when L and R are equal
L > R .... True when L is greater than R ; false when L is less than or equal to R
L < R .... True when L is less than R; false when L is greater than or equal to R
L ≥ R .... True when L is greater than or equal to R; false when L is less than R
L ≤ R .... True when L is less than or equal to R; false when L is greater than R

The following shows how jumps are performed based upon whether or not a condition is true or false.



- Statements following ⇒ and ≠⇒ can be multistatements.
- Up to 15 conditions can be nested inside of a single conditional jump.
- A conditional jump cannot contain any newline symbols (↵). If you include a newline symbol inside of a conditional jump, a **Syn ERROR** will occur when you run the program.

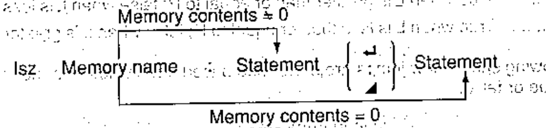
Example **To input a program that calculates the square root of any input value zero or greater and squares any input value that is less than zero.**

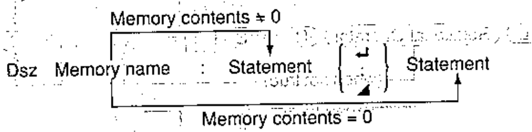


When this program is executed, it first prompts for input of a value for A. If the value of A is 0 or greater, execution jumps to the statement between ⇒ and ◢. If the value of A is less than zero, execution jumps to the statement between ≠ ⇒ and ◢. Finally, an unconditional jump causes execution to jump from Goto 1 to Lbl 1 to repeat the program.

• Count Jumps

There are two count jumps: one that increments a value memory (Isz) and one that decrements a value memory (Dsz). Look at the following format:





As shown above, if the increment or decrement operation does not cause the content of the value memory to become 0, the statement following the value memory name is executed. If the content of the value memory becomes 0, the next statement is skipped.

Example **To write a program that accepts input of 10 values, and then calculates the average of the values.**

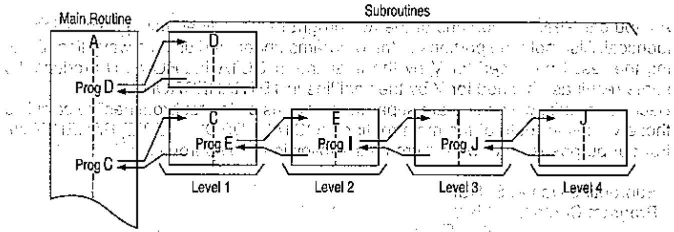
Program

```
A = 10
C = 0
Lbl 1 ◄──────┐
 {B}         │
C = B + C    │ Unconditional jump
Dsz A        │
Goto 1 ──────┘
C ÷ 10
```


This program starts out by assigning a value of 10 to A. This is because value memory A will be used as a control variable. The next statement clears C to zero. After defining the location of label 1 (Lbl 1), the program then prompts for input of a value for B. The next statement adds the value of B to value memory C, and then stores the result in C. The next three statements say: "decrement the value in A, and if it is still greater than 0, jump back to label 1; otherwise divide the contents of C by 10".

■ Subroutines

Up to this point, all of the programs we have seen were contained in a single program area. You can also jump between program areas, so that the resulting execution is made up of pieces in different areas. In such a case, the central program from which other areas are jumped to is called a "main routine". The areas jumped to from the main routine are called "subroutines".



To jump to a subroutine, use Prog (input using [SHIFT] [Prog]) followed by a program file name in double quotation.

Example **Prog ABC — Jumps to the program stored in a file named "ABC".**

After the jump to the program you specify, execution continues from the beginning of the subroutine. When end of the subroutine is reached, execution returns to the statement following the Prog command that initiated the subroutine.

You can jump from one subroutine to another, a procedure that is called "nesting". You can nest up to a maximum of 10 levels, and an error will occur (**Ne ERROR**) if you try to nest an 11th time. If you try to jump to a program area that does not contain a program, an error message (**Go ERROR**) will appear on the display.

Important

- The Goto command does not jump between program areas. A Goto command jumps to the label (Lbl) located inside the same program area.

Example **Create two programs, one that calculates the surface area and volume of a regular octahedron and one that calculates the surface area and volume of a regular tetrahedron. Specify that results should use only three decimal places.**

- Octahedron
  File Name: OCTAHEDRON
  Program Contents: Fix 3
  $S = 2 \times \sqrt{3} \times A^2$ ◢
  $V = \sqrt{2} \div 3 \times A$^3

- Tetrahedron
  File Name: TETRAHEDRON
  Program Contents: Fix 3
  $S = \sqrt{3} \times A^2$ ◢
  $V = \sqrt{2} \div 12 \times A$^3

As you can see, the portions of the two programs underlined using a solid line are identical. Also note the portions of the programs underlined using a wavy line. Dividing the result produced for V by the final line in OCTAHEDRON will produce the same result as obtained for V by the final line in TETRAHEDRON.
Because of this, we can create separate programs called "subroutines" to calculate these values for both of the main routines (OCTAHEDRON and TETRAHEDRON). For our purposes here, we will create the following two subroutines.

Subroutine Name: S. SUB
Program Contents: Fix 3
$S = \sqrt{3} \times A^2$

Subroutine Name: V. SUB
Program Contents: $V = \sqrt{2} \div 3 \times A$^3

Now we can change our main routines to the following.

- Octahedron
  File Name: OCTAHEDRON
  Program Contents: Prog "S. SUB"
  S = Ans × 2 ◢
  Prog "V. SUB"

- Tetrahedron
  File Name: TETRAHEDRON
  Program Contents: Prog "S. SUB" ◢
  Prog "V. SUB"
  V = Ans ÷ 4

The following is a description of each step performed when OCTAHEDRON or TETRAHEDRON is run.

1. When you run either of the two main routines, execution will immediately jump to the subroutine S. SUB.
2. In S. SUB, the Fix 3 command specifies three decimal places.
3. Next, the subroutine calculates the surface area of a regular tetrahedron using the value you input for A.
4. Processing jumps back to the main routine.
   - TETRAHEDRON uses the value calculated by S. SUB as it is.
   - OCTAHEDRON multiplies the result calculated by S. SUB by 2 (S = Ans × 2) to convert it to the surface area of a regular octahedron.
5. The main routine now jumps to V. SUB.
6. The subroutine calculates the volume of a regular octahedron.
7. Processing jumps back to the main routine.
   - OCTAHEDRON uses the value calculated by V. SUB as it is.
   - TETRAHEDRON divides the result calculated by V. SUB by 4 (S = Ans ÷ 4) to convert it to the volume of a regular tetrahedron.

The use of subroutines helps to save memory and makes programs easier to write.

■ Pause Command

The following is the syntax of the Pause command.
Pause *n* (*n* = and integer from 0 to 9)
The Pause command can be used to stop program execution for up to 4.5 seconds. The calculator displays the intermediate result up to that point (Ans Memory contents) while execution is paused.
The following table shows the meaning of the integers used with the Pause command.

n	0	1	2	·····	8	9
Seconds Paused	0	0.5	1	·····	4	4.5

- Pause *n* is treated as one statement.

Example **Write a program that starts A with a value of 1, and then continually increments the value of A, displaying each new value of A for 1.5 seconds.**

```
Lbl 1 ←──────────────────────┐
A = A + 1                    │
Pause 3 (1.5-second display) │  Unconditional jump
Goto 1 ──────────────────────┘
```

In this program A = A + 1 increments the value of A, and Pause 3 displays the new value of A for about 1.5 seconds. The Goto 1/Lbl 1 unconditional jump creates an endless loop.

Program Library

1. Prime Factor Analysis
2. Greatest Common Measure
3. Minimum Loss Matching

CASIO PROGRAM SHEET

Program for	No.
Prime Factor Analysis	1

Description

Produces prime factors of arbitrary positive integers

For $1 < m < 10^{10}$

prime numbers are produced from the lowest value first. "END" is displayed at the end of the program.

(Overview)

m is divided by 2 and by all successive odd numbers (d = 3, 5, 7, 9, 11, 13, ...) to check for divisibility.

Where d is a prime factor, $m_i = m_{i-1}/d$ is assumed, and division is repeated until $\sqrt{m_i} + 1 \leq d$.

Example

[1]
119 = 7 × 17

[2]
630 = 2 × 3 × 3 × 5 × 7

[3]
262701 = 3 × 3 × 17 × 17 × 101

Preparation and operation

- Specify the mode for the program execution.
- Store the program written on the next page.
- Execute the program as shown below.

Step	Key operation	Display	Step	Key operation	Display
1	MODE 1 (COMP) FUNCTION 6 (DSP/CLR) 5 (Mcl) EXE	Mcl 0	11	EXE	PRIME FACTOR= 5
2	FILE	Program [RUN] ■ PRIME FACTOR:CO	12	EXE	PRIME FACTOR= 7
3	EXE	M? 0	13	EXE	END 630
4	119 EXE	PRIME FACTOR= 7	14	EXE	M? 7
5	EXE	PRIME FACTOR= 17	15	262701 EXE	PRIME FACTOR= 3
6	EXE	END 119	16	EXE	PRIME FACTOR= 3
7	EXE	M? 17	17	EXE	PRIME FACTOR= 17
8	630 EXE	PRIME FACTOR= 2	18	EXE	PRIME FACTOR= 17
9	EXE	PRIME FACTOR= 3	19	EXE	PRIME FACTOR= 101
10	EXE	PRIME FACTOR= 3	20	EXE	END 262701

No. 1

Line	(While the "Filename?" screen is on the display, input the following commands.) Program
File name	P R I M E F A C T O R EXE F1 (COMP)
1	Lbl 0 : { A } : A " M " : N = A : Goto 2 ↵
2	Lbl 1 : B " P R I M E F A C T O R " = 2 ◢ A = A ÷ 2 : A = 1 ⇒ Goto 9 : ◺ ↵
3	Lbl 2 : Frac (A ÷ 2) = 0 ⇒ Goto 1 : ◺ B = 3 ↵
4	Lbl 3 : C = √ A + 1 ↵
5	Lbl 4 : B ≥ C ⇒ Goto 8 : ◺ Frac (A ÷ B) = 0 ⇒ Goto 6 : ◺ ↵
6	Lbl 5 : B = B + 2 : Goto 4 ↵
7	Lbl 6 : (A ÷ B) B − A = 0 ⇒ Goto 7 : ◺ Goto 5 ↵
8	Lbl 7 : B " P R I M E F A C T O R = " ◢ A = A ÷ B : Goto 3 ↵
9	Lbl 8 : A " P R I M E F A C T O R = " ◢
10	Lbl 9 : N " E N D " ◢ Goto 0

Memory contents

A	m	H		O		V	
B	d	I		P		W	
C	$\sqrt{m_i}+1$	J		Q		X	
D		K		R		Y	
E		L		S		Z	
F		M		T			
G		N	m	U			

CASIO PROGRAM SHEET

Program for	No.
Greatest Common Measure	2

Description

Euclidean general division is used to determine the greatest common measure for two integers a and b.

For $|a|, |b| < 10^9$, positive values are taken as $< 10^{10}$

(Overview)

$$n_0 = \max(|a|, |b|)$$
$$n_1 = \min(|a|, |b|)$$
$$n_k = n_{k-2} - \left[\frac{n_{k-2}}{n_{k-1}}\right] n_{k-1}$$

$k = 2, 3 \ldots\ldots$

If $n_k = 0$, then the greatest common measure (c) will be n_{k-1}.

Example

	[1]	[2]	[3]
When	$a = 238$	$a = 23345$	$a = 522952$
	$b = 374$	$b = 9135$	$b = 3208137866$
	↓	↓	↓
	$c = 34$	$c = 1015$	$c = 998$

Preparation and operation

- Specify the mode for the program execution.
- Store the program written on the next page.
- Execute the program as shown below.

Step	Key operation	Display
1	MODE 1 (COMP) FUNCTION 6 (DSP/CLR) 5 (Mcl) EXE	Mcl 0
2	FILE	Program [RUN] ■PRIME FACTOR:CO MEASURE :CO
3	▼ EXE	A? 0
4	238 EXE	B? 0
5	374 EXE	C 34
6	EXE	A? 102
7	23345 EXE	B? 34
8	9135 EXE	C 1015
9	EXE	A? 4060
10	522952 EXE	B? 1015
11	3208137866 EXE	C 998

No.
2

Line / (While the "Filename?" screen is on the display, input the following commands.) **Program**

Line																		
File name	M	E	A	S	U	R	E	EXE	1	(COMP)								
1	Lbl	1	↵															
2	{	A	,	B	}	↵												
3	A	=	Abs	A	:	B	=	Abs	B	↵								
4	B	>	A	⇒	C	=	A	:	A	=	B	:	B	=	C	:	◺	↵
5	Lbl	2	↵															
6	C	=	(−)	(	Int	(	A	÷	B	)	×	B	−	A	)	↵		
7	C	≠	0	⇒	A	=	B	:	B	=	C	:	Goto	2	:	◺	↵	
8	B	"	C	"	◢	Goto	1											

Memory contents

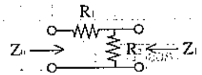
A	a, n_0	H		O		V	
B	b, n_1	I		P		W	
C	n	J		Q		X	
D		K		R		Y	
E		L		S		Z	
F		M		T			
G		N		U			

CASIO PROGRAM SHEET

Program for	No.
Minimum Loss Matching	3

Description

Calculate R_1 and R_2 whih match Z_0 and Z_1 with loss minimized. $(Z_0 > Z_1)$



$$R_1 = Z_0\sqrt{1-\frac{Z_1}{Z_0}} \qquad R_2 = \frac{Z_1}{\sqrt{1-\frac{Z_1}{Z_0}}}$$

$$\text{Minimum loss } L_{min} = 20\cdot\log\left(\sqrt{\frac{Z_0}{Z_1}} + \sqrt{\frac{Z_0}{Z_1}-1}\right) \text{[dB]}$$

Example

Calculate the values of R_1, R_2 and L_{min} for $Z_0 = 500\Omega$ and $Z_1 = 200\Omega$.

Preparation and operation

- Specify the mode for the program execution.
- Store the program written on the next page.
- Execute the program as shown below.

Step	Key operation	Display	Step	Key operation	Display
1	MODE 1 (COMP) FUNCTION 6 (DSP/CLR) 5 (Mcl) EXE	Mcl 0			
2	FILE	Program [RUN] ■PRIME FACTOR :CO MEASURE :CO LOSS :CO			
3	▼ ▼ EXE	Z0 ? 0			
4	500 EXE	Z1? 0			
5	200 EXE	R1 = 387.2983346			
6	EXE	R2 = 258.1988897			
7	EXE	LMIN = 8.961393328			

No.
3

Line	(While the "Filename?" screen is on the display, input the following commands.) Program
File name	L O S S EXE I (COMP)
1	Y " Z 0 " : Z " Z 1 " ↵
2	A = √ (1 − Z ÷ Y) : B = Y ÷ Z ↵
3	R " R 1 " = Y A ◢
4	S " R 2 " = Z ÷ A ◢
5	T " L M I N " = 2 0 log (√ B + √ (B
	− 1))

Memory contents

A	$\sqrt{1-\frac{Z_1}{Z_0}}$	H		O		V	
B	$\frac{Z_0}{Z_1}$	I		P		W	
C		J		Q		X	
D		K		R	R_1	Y	Z_0
E		L		S	R_2	Z	Z_1
F		M		T	L_{min}		
G		N		U			

Appendix

Appendix A Error Message Table

Message	Meaning	Countermeasure
Syn ERROR	① Calculation formula contains an error. ② Formula in a program contains an error.	① Use [◀] or [▶] to display the point where the error was generated and correct it. ② Use [◀] or [▶] to display the point where the error was generated and then correct the program.
Ma ERROR	① Calculation result exceeds calculation range. ② Calculation is performed outside the input range of a function. ③ Illogical operation (division by zero, etc.)	①②③ Check the input numeric value and correct it. When using variables, check that the numeric values assigned to variables are correct.
Go ERROR	① No corresponding Lbl *n* for Goto *n*. ② No program stored in program area Prog "file name".	① Correctly input a Lbl *n* to correspond to the Goto *n*, or delete the Goto *n* if not required. ② Store a program in program area Prog "file name", or delete the Prog "file name" if not required.
Ne ERROR	• Nesting of subroutines by Prog "file name" exceeds 10 levels.	• Ensure that Prog "file name" is not used to return from subroutines to main routine. If used, delete any unnecessary Prog "file name". • Trace the subroutine jump destinations and ensure that no jumps are made back to the original program area. Ensure that returns are made correctly.
Stk ERROR	• Execution of calculations that exceed the capacity of the stack for numeric values or stack for calculations. • Program being run contains a conditional jump that contains more than 15 conditions.	• Simplify the formulas to keep stacks within 10 levels for the numeric values and 26 levels for the calculations. • Divide the formula into two or more parts.

Mem ERROR	• Specified expanded value memory does not exist.	• Use [SHIFT] [Defm] to correctly expand the number of value memories.
Arg ERROR	① Incorrect argument specification for a command that requires an argument. ② Not enough memory to expand value memories specified number.	① Correct the argument. • Sci *n*, Fix *n*: *n* = integer from 0 through 9. • Lbl *n*, Goto *n*: *n* = integer from 0 through 9, A through Z. • Defm *n*: *n* = integer from 0 up to the number of remaining bytes. ② • Keep the number of value memories you use for the operation within the number of value memories currently available. • Simplify the data you are trying to store to keep it within the available memory capacity. • Delete no longer needed data to make room for the new data.

Appendix B Input Ranges

Function	Input ranges	Internal digits	Accuracy	Notes
sin$x$ cos$x$ tan$x$	(DEG) $\lvert x\rvert < 9 \times 10^{9}$° (RAD) $\lvert x\rvert < 5 \times 10^{7}\pi$rad (GRA) $\lvert x\rvert < 1 \times 10^{10}$grad	15 digits	As a rule, accuracy is ±1 at the 10th digit.	However, for tan$x$: $\lvert x\rvert \neq 90(2n+1)$:DEG $\lvert x\rvert \neq \pi/2(2n+1)$:RAD $\lvert x\rvert \neq 100(2n+1)$:GRA
$\sin^{-1}x$ $\cos^{-1}x$	$\lvert x\rvert \leqq 1$	"	"	
$\tan^{-1}x$	$\lvert x\rvert < 1 \times 10^{100}$			
sinh$x$ cosh$x$	$\lvert x\rvert \leqq 230.2585092$	"	"	Note: For sinh and tanh, when x=0, errors are cumulative and accuracy is affected at a certain point.
tanh$x$	$\lvert x\rvert < 1 \times 10^{100}$			
$\sinh^{-1}x$	$\lvert x\rvert < 5 \times 10^{99}$	"	"	
$\cosh^{-1}x$	$1 \leqq x < 5 \times 10^{99}$			
$\tanh^{-1}x$	$\lvert x\rvert < 1$			
log$x$ ln$x$	$1 \times 10^{-99} \leqq x < 1 \times 10^{100}$	"	"	
10^x	$-1 \times 10^{100} < x < 100$	"	"	
e^x	$-1 \times 10^{100} < x \leqq 230.2585092$			
$\sqrt{x}$	$0 \leqq x < 1 \times 10^{100}$	"	"	
x^2	$\lvert x\rvert < 1 \times 10^{50}$			
x^{-1} (1/x)	$\lvert x\rvert < 1 \times 10^{100}$, $x \neq 0$	"	"	
$\sqrt[3]{x}$	$\lvert x\rvert < 1 \times 10^{100}$			
$x!$	$0 \leqq x \leqq 69$ (x is an integer)	"	"	
nPr nCr	Result < 1×10^{100} n, r (n and r are integers) $0 \leqq r \leqq n$, $n < 1 \times 10^{10}$	"	"	
Pol (x, y)	$\sqrt{x^2 + y^2} < 1 \times 10^{100}$	"	"	

Function	Input ranges	Internal digits	Accuracy	Notes
Rec (r, θ)	$\lvert r\rvert < 1 \times 10^{100}$ (DEG) $\lvert\theta\rvert < 9 \times 10^{9}$° (RAD) $\lvert\theta\rvert < 5 \times 10^{7}\pi$ rad (GRA) $\lvert\theta\rvert < 1 \times 10^{10}$grad	15 digits	As a rule, accuracy is ±1 at the 10th digit.	However, for tanθ: $\lvert\theta\rvert \neq 90(2n+1)$:DEG $\lvert\theta\rvert \neq \pi/2(2n+1)$:RAD $\lvert\theta\rvert \neq 100(2n+1)$:GRA
° ' "	$\lvert a\rvert$, b, $c < 1 \times 10^{100}$ $0 \leqq b, c$	"	"	
$\overleftarrow{° ' "}$	$\lvert x\rvert < 1 \times 10^{100}$ Hexadecimal display: $\lvert x\rvert \leqq 2777777.777$			
^ (x^y)	$x > 0$: $-1 \times 10^{100} < y\log x < 100$ $x = 0$: $y>0$ $x < 0$: $y=n$, $\frac{1}{2n+1}$ (n is an integer) However; $-1 \times 10^{100} < \frac{1}{y}\log\lvert x\rvert < 100$	"	"	
$\sqrt[x]{y}$	$y > 0$: $x \neq 0$ $-1 \times 10^{100} < \frac{1}{x}\log y < 100$ $y=0$: $x > 0$ $y < 0$: $x = 2n+1$, $\frac{1}{n}$ ($n \neq 0$, n is an integer) However; $-1 \times 10^{100} < \frac{1}{x}\log\lvert y\rvert < 100$	"	"	
a^b/c	• Results Total of integer, numerator and denominator must be within 10 digits (includes division symbols). • Input Result displayed as a fraction for integer when integer, numerator and denominator are less than 1×10^{10}.	"	"	
SD (LR)	$\lvert x\rvert < 1 \times 10^{50}$ $\lvert y\rvert < 1 \times 10^{50}$ $\lvert n\rvert < 1 \times 10^{100}$ $x\sigma_n$, $y\sigma_n$, $\bar{x}$, $\bar{y}$, A, B, r: $n \neq 0$ $x\sigma_{n-1}$, $y\sigma_{n-1}$: $n \neq 0, 1$	"	"	

Function	Input Range
BASE-N	Values after variable within following range: DEC: $-2147483648 \leqq x \leqq 2147483647$ BIN: $10000000000000000000000000000000 \leqq x$ $\leqq 11111111111111111111111111111111$ (negative) $0 \leqq x \leqq 01111111111111111111111111111111$ (0, positive) OCT: $20000000000 \leqq x \leqq 37777777777$ (negative) $0 \leqq x \leqq 17777777777$ (0, positive) HEX: $80000000 \leqq x \leqq$ FFFFFFFF (negative) $0 \leqq x \leqq$ 7FFFFFFF (0, positive)

* Errors may be cumulative with internal continuous calculations such as $\wedge(x^y)$, $\sqrt[x]{y}$, $x!$, $\sqrt[3]{x}$ sometimes affecting accuracy.

Appendix C Specifications

Model: *fx*-4800P

Calculations

Basic calculation functions:
Negative numbers; exponents; parenthetical addition, subtraction, multiplication, division (with priority sequence judgement function - true algebraic logic)

Built-in scientific functions:
Trigonometric/inverse trigonometric functions (units of angular measurement: degrees, radians, grads); hyperbolic/inverse hyperbolic functions; logarithmic/exponential functions; reciprocals; factorials; square roots; cube roots; powers; roots; squares; negative signing; exponential notation input; π; parenthetical calculations; internal rounding; random numbers; unit of angular measurement specification; fractions; decimal-sexagesimal conversion; coordinate transformation; engineering calculations; permutation; combination; number of decimal place and significant digit specification

Built-in functions:
Exponential notation range; delete, insert, answer functions; replay; memory status display; multistatements; output command

Differentials:
Extraction of derivative using differential from center point

Quadratic Differentials:
Extraction of quadratic differential using second order value differential formula

Integrations:
Using Simpson's rule

Σ Calculations:
Calculation of partial sum of sequence $\{a_n\}$

Complex Number Calculations:
Addition, subtraction, multiplication, division; reciprocal; square root; square; absolute number/argument calculations; conjugate complex number extraction; real number part/imaginary number part extraction

Binary, Octal, Decimal, Hexadecimal Calculations:
Addition, subtraction, multiplication, division calculations; base specification; negative values (two's complement); logical operations

Statistics:

Standard deviation: number of data; mean; standard deviation (two types); sum; sum of squares; t-test

Regression: number of data; mean of x; mean of y; standard deviation of x (two types); standard deviation of y (two types); sum of x; sum of y; sum of squares of x; sum of squares of y; sum of squares of x and y; constant term; regression coefficient; correlation coefficient; estimated value of x; estimated value of y

Formula Memory:
Formula storage, recall, execution; Table function; Solve function; storage of formula to program area, recall

Variables:
26 standard (expandable to 476)

Calculation Range:
$\pm 1 \times 10^{-99}$ to $\pm 9.999999999 \times 10^{99}$ and 0. Internal operations use 15-digit mantissa.

Exponential Display:
Norm 1: $10^{-2} > |x|, |x| \geqq 10^{10}$
Norm 2: $10^{-9} > |x|, |x| \geqq 10^{10}$

Rounding:
Performed according to the number of significant digits and number of decimal places specified by user.

Programming

Programming:
Input, storage, recall, execution of programs in program area; editing and deletion of file names and program contents; recall by file name

Program commands:
Variable input ({}); Variable lock (Fixm); Unconditional jump (Goto, Lbl); Conditional jump ⇒, ≒ ⇒; ◢; relational operators (=, ≒, >, <, ≥, ≤); Count jump (Isz, Dsz); Subroutine (Prog) with nesting up to 10 levels; Pause (Pause)

Check function:
Program check; debugging

Program area:
4,500 bytes maximum

General

Display system:
16-character × 4-line liquid crystal display; 10-digit mantissa and 2-digit exponent for calculations: displays binary, octal, hexadecimal, sexagesimal, fraction values, complex numbers

Text display:
Up to 64 characters for function commands, program commands, alpha characters

Error check function:
Check for illegal calculations (using values greater than 10^{100}), illegal jumps, etc. Indicates by error message display.

Power supply:
Main: One CR2032 lithium battery
Back-up: One CR2032 lithium battery

Power consumption:
0.05W

Battery life:
Main: Approximately 900 hours (continuous display of 0.)
Approximately 1 year (power off)
Back-up: Approximately 2 years

Auto power off:
Power is automatically turned off approximately six minutes after last operation.

Ambient temperature range:
0°C to 40°C

Dimensions:
Closed: 15mmH × 81.5mmW × 157mm D (5/8"H × 3 1/4"W × 6 1/8"D)
Open: 11mmH × 165mmW × 157mm D (3/8"H × 6 1/2"W × 6 1/8"D)
Weight: 133g (4.7oz) (including batteries)