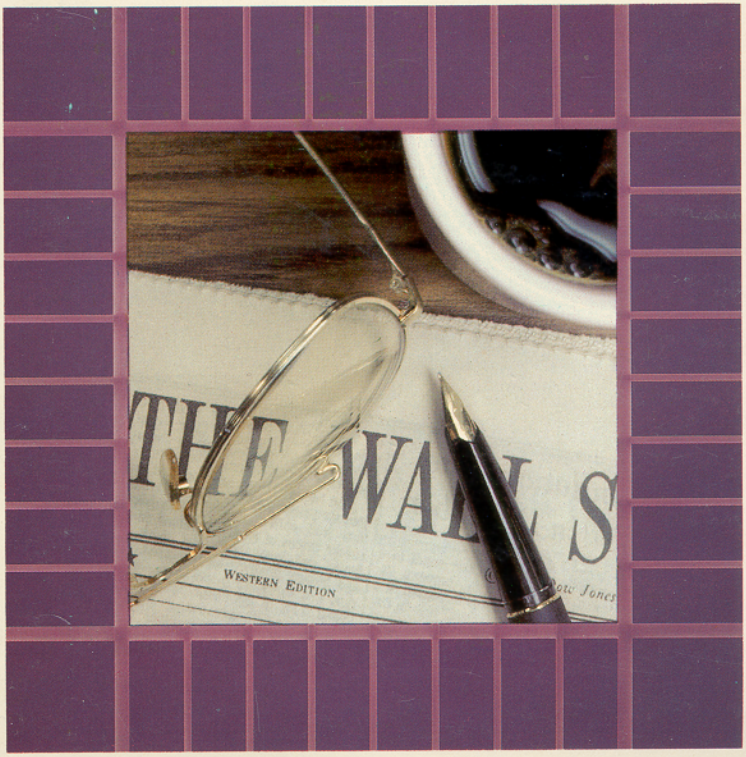
HEWLETT-PACKARD

HP-38E/38C

OWNER'S HANDBOOK AND PROGRAMMING GUIDE



About Continuous Memory (HP-38C only)

If you purchased an HP-38C instead of an HP-38E, your calculator contains Continuous Memory—one of the most advanced memory systems available in a financial calculator. Even when the calculator is turned off, the financial registers and the storage registers save information such as cash flows, amortization data, or intermediate answers for later updating or calculations. Continuous Memory also allows you to save any programs you frequently find useful, whether short, like a checkbook balancer, or longer, like a bond calculation, for as long as you want.

And, continuous memory helps save battery life! Much of the time a calculator is switched on is spent holding information, waiting for the next computation. Continuous Memory eliminates much of this unneeded battery usage by allowing you to switch off the calculator when you have a prolonged pause or interruption between calculations.

Note: Unless otherwise indicated, "HP-38E" is used throughout this handbook to refer to both the HP-38E and the HP-38C.

HP-38E/38C
Owner's Handbook and Programming Guide

August 1980

00038-90038 Rev. B

Printed in U.S.A.

© Hewlett-Packard Company 1980

Contents

The HP-38E/38C Financial Programmable Calculator

The HP-38E/38C Memory 6
Function and Key Index 6
Programming Key Index 10

A Financial Calculator With Programming Capability! 12

Manual Problem Solving 12
Programmed Problem Solving 13

Part I: Problem Solving 17

Section 1: Using Your Calculator 18

The Keyboard 18
Clearing Prefixes 18
Arithmetic Operators [+] [−] [×] [÷] 19
Percentage [%] 19
Net Amount [%] [+] or [%] [−] 19
Percent Difference Between Two Numbers [Δ%] 20
Percent of Total [%T] 21
LAST X 22
The Hidden Function (Mantissa) 22
Keying In Exponents of 10 23
Calendar 24
Date Format 24
Days Between Dates [ΔDAYS] 24
Future or Past Dates [DATE] 25

Section 2: The Financial Functions 28

The Financial Registers 28
Displaying Financial Values 29
Clearing the Financial Registers 29
Reentering Values 29
The Payment Switch 29
Financial Interest Calculations 30
Simple Interest [INT] 32
Accrued Interest—360-Day Basis 32
Accrued Interest—365-Day Basis 34

Compound Interest 34
Solving for the Number of Compounding or Payment Periods [n] 34
Solving for the Periodic Interest Rate [i] 37
Solving for the Present Value [PV] 39
Solving for the Periodic Payment Amount [PMT] 42
Solving for the Future Value [FV] 43
Amortization [AMORT] 45
Discounted Cash Flow Analysis 48
Net Present Value [NPV] 49
Internal Rate of Return [IRR] 53
Changing Cash Flow Entries 56
Reviewing Cash Flow Entries 56

Section 3: Mathematical Functions 58
Reciprocals [1/x] 58
Square Root [√x] 58
Exponentiation: Raising a Number to a Power [y^x] 59
Logarithms [LN] 61
Antilogarithms [e^x] 61
Factorials [n!] 62
Number Alteration 63
Rounding a Number [RND] 63
Integer Portion of a Number [INTGR] 64
Fractional Portion of a Number [FRAC] 65

Section 4: Statistical Functions 66
Summations 66
Viewing the Statistical Registers 67
Deleting and Correcting Data 67
Mean [x̄] 67
Standard Deviation [s] 69
Weighted Mean [x̄w] 70
Linear Regression 72
Correlation Coefficient 72
Linear Estimates x̂, ŷ 74

Part II: Programming 79

Section 5: What Is a Program? 80
Why Write Programs? 80
Three Modes of Operation 81
Introductory Program 82
Keycodes 82
Running a Program 84

GTO 00 84
Automatic Memory Allocation 85

Section 6: Writing a Second Program 90
Displaying Each Step 93
Displaying a Particular Step 93
Interrupting Program Execution 94
Stopping Program Execution 94
Pausing During Program Execution 97
Program Stops 99

Section 7: Decisions 102
Flowcharts 102
Branching 106
Unconditional Branching and Looping 106
Conditional Tests and Conditional Branching 108

Section 8: Editing a Program 118
Finding the Error 118
SST Execution 118
Cued Stops 120
Changing One Instruction 120
Adding Instructions 122
Multiple Programs 124

Appendix A: Service and Maintenance 126

Appendix B: More About IRR 136

Appendix C: Error Messages 138

Appendix D: Continuous Memory 140

Index 141

The HP-38E/38C Financial Programmable Calculator

Fold Out

Financial Registers

n
i
PV
PMT
FV

N_0
N_1
N_2
N_{20}

Automatic Memory Stack

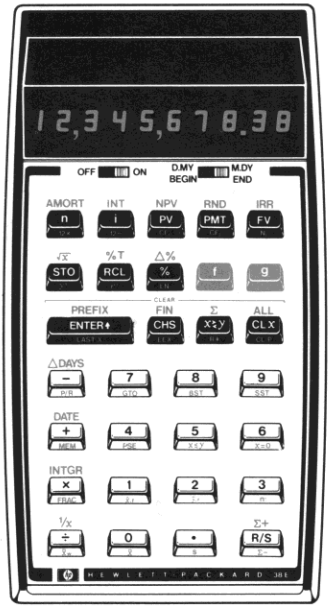
Registers

T
Z
Y
X

LAST X

Storage Registers

R_0			$R_{.0}$
R_1		n	$R_{.1}$
R_2		Σx	$R_{.2}$
R_3		Σx^2	$R_{.3}$
R_4		Σy	$R_{.4}$
R_5		Σy^2	$R_{.5}$
R_6		Σxy	$R_{.6}$
R_7			$R_{.7}$
R_8			$R_{.8}$
R_9			$R_{.9}$



Program Memory

00	09
01	10
02	11
03	⋮
04	
05	
06	97
07	98
08	99

Converted from storage registers.

The calculator automatically converts one storage register into seven lines of program memory, one register at a time as you need them, beginning with $R_{.9}$ and ending with R_7.

Storage register arithmetic can be performed in storage registers R_1 through R_6. Refer to *Your HP Financial Calculator* for instructions on how to use storage register arithmetic.

Function and Key Index

OFF ON

Power switch **(page 13).**

D.MY BEGIN M.DY END

Payment switch. Selects timing of payments in compound interest calculations **(page 29)** or selects date format **(page 24).**

[f] Shift key. Selects alternate function in gold above the function keys. Also used in display formatting **(page 18).**

[g] Shift key. Selects alternate function in blue on the slanted face of the function keys **(page 18).**

CLEAR [PREFIX] after [f], [g], [STO], [RCL], or [GTO], cancels that key **(page 18).**

[f] CLEAR [PREFIX] is also the mantissa function **(page 22).**

Digit Entry

[ENTER↑] Enters a copy of number in displayed X-register into Y-register. Used to separate numbers.*

[CHS] Changes sign of number or exponent of 10 in displayed X-register.*

[EEX] Enter exponent. After pressing, next numbers keyed in are exponents of 10 **(page 23).**

[0] thru [9] Digits. Used for keying in numbers and display formatting.*

[.] Decimal point. Also used for display formatting.*

[CLx] Clears contents of displayed X-register to zero.*

Reviewing Numbers

[x≷y] Exchanges contents of X- and Y-registers of stack.*

[R↓] Rolls down contents of stack for viewing in displayed X-register.*

[LAST x] Recalls number displayed before the previous operation back into the displayed X-register **(page 22).**

Manual Storage

[STO] Store. Followed by number key, decimal point and number key, or top row financial key, stores displayed number in storage register specified. Also used to perform storage register arithmetic.*

* Refer to *Your HP Financial Calculator*.

[RCL] Recall. Followed by number key, decimal point and number key, or top-row financial key recalls value from storage register specified into the displayed X-register.*

CLEAR [ALL] Clears contents of stack (X,Y,Z,T), all storage registers, statistical registers, and financial registers. Leaves program memory untouched.*

Arithmetic

[+] [−] [×] [÷] Arithmetic operators **(page 19).***

Percentages

[%] Computes x% of y and retains the y value in the Y-register **(page 19).***

[Δ%] Computes percent of change between number in Y-register and number in displayed X-register **(page 20).***

[%T] Computes percent that x is of number in Y-register **(page 21).***

Calendar

[DATE] Changes a date in the Y-register by the number of days in the X-register and displays day of week **(page 25).**

[ΔDAYS] Computes the number of days between two dates in the Y- and X-registers **(page 24).**

Compound Interest

CLEAR [FIN] Clears contents of financial registers **(page 29).**

[n] Stores or computes number of periods in financial problem **(page 34).**

[12×] Multiplies displayed X-register by 12 and stores the resulting value in the n-register **(page 28).**

[i] Stores or computes interest rate per compounding period **(page 37).**

[12÷] Divides displayed X-register by 12 and stores the resulting value in the i-register **(page 28).**

[PV] Stores or computes present value (initial cash flow) of a financial problem **(page 39).**

[PMT] Stores or computes payment amount **(page 42).**

[FV] Stores or computes future value (final cash flow) of a financial problem **(page 43).**

[AMORT] Amortizes x number of periods using values stored in [PMT], [i], [PV], and the display. Updates values in [PV] and [n] **(page 45).**

[INT] Calculates simple interest **(page 32).**

[NPV] Calculates the net present value of up to 20 uneven cash flows and initial investment using values stored with [CF_0], [CF_j], and [N_j] **(page 49).**

---

* Refer to *Your HP Financial Calculator*.

[IRR] Calculates the internal rate of return (yield) for up to 20 uneven cash flows and initial investment using values stored with [CF_0], [CF_j], and [N_j] **(page 53).**

[CF_0] Initial cash flow. Stores contents of displayed X-register in R_0, initializes [n] to zero, starts automatic IRR, sets N_0 to 1. Used at the beginning of a discounted cash flow problem. **(page 49).**

[CF_j] Cash flow j. Stores contents of displayed X-register in R_j, increments [n] by 1, sets N_j to 1. Used for all cash flows except the initial cash flow in a discounted cash flow problem **(page 49).**

[N_j] Stores the number (from 1 to 99) of times each cash flow occurs in N_j. Assumes 1 unless otherwise specified **(page 49).**

Mathematics

[$\sqrt{x}$] Computes square root of number in displayed X-register **(page 58).**

[y^x] Raises number in Y-register to power of number in X-register **(page 59).**

[$1/x$] Computes reciprocal of number in displayed X-register **(page 58).**

[n!] Computes factorial $[n \cdot (n-1)... 3 \cdot 2 \cdot 1]$ of number in displayed X-register **(page 62).**

[e^x] Natural antilogarithm. Raises e (approx. 2.718281828) to power of number in displayed X-register **(page 61).**

[LN] Computes natural logarithm (base e, approx. 2.718281828) of number in displayed X-register **(page 61).**

Number Alteration

[RND] Rounds mantissa of 10-digit number in X-register to match the display setting. **(page 63).**

[INTGR] Leaves only the integer portion of number in displayed X-register by truncating fractional portion **(page 64).**

[FRAC] Leaves only the fractional portion of number in displayed X-register by truncating integer portion **(page 65).**

Statistics

CLEAR [Σ] Clears statistical storage registers R_1 through R_6 to zeros **(page 67).**

[$\Sigma+$] Accumulates numbers from X- and Y-registers into statistical storage registers R_1 through R_6 **(page 66).**

[$\Sigma-$] Subtracts x and y values from statistical storage registers R_1 through R_6 for correcting [$\Sigma+$] accumulations **(page 67).**

[$\bar{x}$] Computes mean (average) of x and y values accumulated by [$\Sigma+$] **(page 67).**

$\bar{x}_w$ Computes weighted average of y (item) and x (weight) values accumulated by $\Sigma+$ **(page 70).**

s Computes sample standard deviations of x and y values accumulated by $\Sigma+$ **(page 69).**

$\hat{y},r$ Linear estimate (X-register), correlation coefficient (Y-register). Fits a line to a set of x,y data points entered using $\Sigma+$, then extrapolates this line to estimate a y value ($\hat{y}$) for a given x value. Also computes strength of linear relationship (r) among that set of x,y data points **(page 72).**

$\hat{x},r$ Linear estimate (X-register), correlation coefficient (Y-register). Fits a line to a set of data points entered using $\Sigma+$, then extrapolates this line to estimate an x value ($\hat{x}$) for a given y value. Also computes strength of linear relationship (r) among that set of x,y data points **(page 72).**

Programming Key Index

[P/R] Program/Run. Toggles into and out of *program* mode. Automatically sets program to line 00 when returning to *run* mode **(page 81).**

[MEM] Memory map. Describes the current allocation of memory; the number of lines allotted to program memory and the number of available data registers **(page 87).**

Program Mode	**Automatic Run Mode**	
In *program* mode, function keys are recorded in program memory. Display shows program memory line number and the keycode (keyboard row and location in row) of the function key.	In automatic *run* mode, function keys may be executed as part of a recorded program or individually by pressing from the keyboard.	
Active keys: In program mode only the following keys are active; they cannot be recorded in program memory.	**Pressed from keyboard:**	**Executed as a recorded program instruction:**
[CL P] Clear program. Clears program memory to all [GTO] 00 instructions and resets calculator so operations begin at line 00 of program memory. Resets [MEM] to *P-08 r-20* **(page 82).**	[CL P] Resets calculator (in run mode) so operations begin at line 00 of program memory. Does not erase program memory **(page 84).** [R/S] Run/Stop. Begins execution of a stored program. Stops execution if program is running **(page 84).**	[R/S] Run/Stop. Stops program execution **(page 94).**

Program Mode	Automatic Run Mode	
Active keys:	**Pressed from keyboard:**	**Executed as a recorded program instruction:**
[GTO] Go to. Followed by a decimal point and a two-digit number, positions calculator to that step number of program memory. No instructions are executed **(page 93).**	[GTO] Go to. Followed by a two-digit number, positions calculator to that step number of program memory. No instructions are executed **(page 93).**	[GTO] Go to. Followed by a two-digit number, causes calculator to branch to the specified line number next, and resumes program execution from there **(page 106).**
[SST] Single step. Displays line number and contents of next program memory line **(page 93).**	[SST] Single step. Displays line number and keycode of current program memory line when pressed; executes instruction, displays result, and moves to next line when released **(page 118).**	[PSE] Pause. Stops program execution for about 1 second and displays contents of X-register then resumes program execution **(page 97).**
[BST] Back step. Displays line number and contents of previous program memory line. When back stepped from line 00, goes to end of program memory as defined by [9] [MEM] **(page 93).**	[BST] Back step. Displays line number and keycode of previous program memory line when pressed; displays original contents of X-register when released. No instructions are executed. **(page 119).**	[x≤y] [x=0] Conditional. [x≤y] tests value in X-register against that in Y-register. [x=0] tests value in X-register against zero. If true, calculator executes next program memory line. If false, calculator skips next line before resuming execution **(page 109).**
	Any key. Pressing any key on the keyboard stops execution of a program.	

A Financial Calculator With Programming Capability!

Your HP-38E is a versatile, handheld financial calculator that uses the powerful Hewlett-Packard logic system to compute answers to complex financial, statistical, and mathematical problems in either of two modes:

- **Manual Problem Solving.** You work step-by-step through the toughest of problems, choosing from among the dozens of available operations to calculate the correct answer quickly and easily.
- **Programmed Problem Solving.** The HP-38E memorizes a sequence of up to 99 different operations as you press them, and then repeats that sequence at the touch of one key as often as you wish to solve a particular type of problem.

A *program* is nothing more than a sequence of manual keystrokes that is remembered by the calculator. You can then *execute* the program as often as you like. No prior computer programming experience is necessary for HP-38E calculator programming.

To see the close relationship between the manual solution to a problem and a programmed solution, let's solve a problem manually, and then use a program to solve the same problem and others like it.

Manual Problem Solving

Example: The furniture store in your area is having a huge clearance sale: 25% off all marked prices (before taxes). If the sales tax is 6%, how much does the sofa and two matching armchairs cost, now marked $1,780 for the set?

First, slide the calculator power switch OFF ON to ON. To use the HP-38E to manually compute the discounted cost of the sofa and armchairs, press the following keys in the order shown:

Keystrokes	Display	
1780 [ENTER↑]	*1,780.00*	Enter marked price.
25 [%]	*445.00*	Amount of discount (25% of $1,780).
[−]	*1,335.00*	Price less discount.
6 [%]	*80.10*	Sales tax.
[+]	*1,415.10*	Total cost.

Programmed Problem Solving

If you want to compute the discounted cost plus tax for each piece or set of furniture you are interested in, you could repeat the above procedure, entering a different marked price each time. However, you might wish to write a program that would calculate the discounted price plus tax, instead of pressing all the keys for each item.

To calculate the discounted price plus sales tax using a program, you should first *write* the program, then you must *record* the program into the calculator, and finally you *run* the program to calculate the answer.

Writing the Program: *You have already written it!* A program is nothing more than the series of keystrokes you would execute to solve the same problem manually.

Recording the Program: To record the keystrokes of the program into the calculator:

1. Press [9] [P/R] *(program/run)* to put the calculator in *program* mode.
2. Press [9] [CL P] *(clear program)* to clear the calculator of previous programs.

3. Press the following keys in order. (When you are recording a program, the display gives you information that you will find useful later, but you can ignore the display for now.)

[ENTER↑]
2
5
[%]
[−]
6
[%]
[+]

These keys are the same keys you pressed to solve the problem manually.

Running the Program: Press [9] [P/R] again to get back into *run* mode. Now all you have to do to calculate the discounted cost of any piece of furniture is key in the marked price and press the [R/S] *(run/stop)* key. When you press [R/S], the sequence of keystrokes you recorded is automatically executed by the calculator, giving you the same answer you would have obtained manually.

For example, to calculate the discounted cost plus sales tax of the sofa and armchair set:

Keystrokes	Display	
1780	*1,780.*	
[R/S]	*1,415.10*	Discounted cost plus 6% sales tax.

For the dinette set, marked at $950, and the TV/stereo home entertainment center, marked at $1,925:

Keystrokes	Display	
950 [R/S]	*755.25*	Discounted cost plus 6% sales tax.
1925 [R/S]	*1,530.38*	Discounted cost plus 6% sales tax.

Programming the HP-38E is *that* easy! The calculator remembers a series of keystrokes and then executes them when you press the R/S key.

If you are new to HP calculators and have not yet read *Your HP Financial Calculator: An Introduction to Financial Concepts and Problem Solving*, please do so now. We want you to feel comfortable with your HP-38E and we want to show you how easy it is to use.

Part I of your *HP-38E Owner's Handbook and Programming Guide* reviews the function keys on your keyboard used in manual or programmed problem solving while part II is a complete guide to HP-38E calculator programming.

PART I
Problem Solving

Section 1

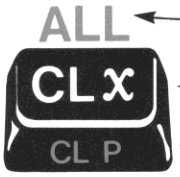
Using Your Calculator

The following section will briefly review arithmetic and percentage calculations, as well as introduce some special features on your HP-38E. If you are already familiar with these functions, turn immediately to page 22.

The Keyboard

Most keys on the HP-38E perform two or three functions. One function is indicated by the symbol on the key face, the second function is indicated by the gold symbol above the key, and the third function is indicated in blue on the slanted face of the key.

- To select the function on the face of the key, press the key.
- To select the function printed in gold above the key, press the gold prefix key [f] then press the function key.
- To select the function printed in blue on the slanted face of the key, press the blue prefix key [g] then press the function key.



To execute the *clear all* function, press [f] CLEAR [ALL].

To execute the *clear x* function, press [CLx].

To execute the *clear program* function, press [g] [CL P].

In this handbook, the selected key function will appear in the appropriate color outlined by a box, like this: CLEAR [ALL], [CLx], [CL P].

Clearing Prefixes

If you make a mistake while keying in a prefix key for a function, press [f] CLEAR [PREFIX]. When you press [f] CLEAR [PREFIX] after [f], [g], [STO], [RCL], or [GTO], that prefix key will be cancelled and all 10 digits in the display will be shown for approximately one second.

Arithmetic Operators [+] [−] [×] [÷]

To perform an arithmetic operation:

1. Key in the first number.
2. Press [ENTER↑] to separate the first number from the second.
3. Key in the second number.
4. Press the proper key [+], [−], [×], or [÷] to perform the desired arithmetic operation.

Solve	Keystrokes	Display
10 ÷ 5	10 [ENTER↑] 5 [÷]	*2.00*
10 + 1 + 2 + 3	10 [ENTER↑] 1 [+]	
	2 [+] 3 [+]	*16.00*

Refer to *Your HP Financial Calculator* for further information concerning simple arithmetic, chain arithmetic, and the order of execution.

Percentage [%]

To find the percentage of a number, key in that base number and press [ENTER↑]. Then key in the numerical value of the percent and press [%].

Calculate	Keystrokes	Display
13% of 2,500	2500 [ENTER↑]	*2,500.00*
	13 [%]	*325.00*

Notice that the keystroke sequence is similar to the keystrokes you learned for arithmetic operations. The function happens immediately when you press the [%] key.

Net Amount [%][+] or [%][−]

It is easy to calculate the net amount because the calculator automatically holds the base number while you calculate percentages. To find the net amount, calculate the percentage amount and then press [+] or [−] depending on whether you wish to add or subtract that amount from the base number.

Example: If the sales tax on a $6,200 car is 5%, what is the amount of the tax and total cost of the car?

Keystrokes	Display	
6200 [ENTER↑]	*6,200.00*	Enter base amount.
5 [%]	*310.00*	Percentage amount (sales tax).
[+]	*6,510.00*	Net amount (base plus percentage amount).

Percent Difference Between Two Numbers [Δ%]

To find the percent difference between two numbers, i.e., the ratio of increase or decrease:

1. Key in the first number (old value).
2. Press [ENTER↑] to separate the first number from the second.
3. Key in the second number (new value).
4. Press [f][Δ%]

Example: Your rent jumps from $285 a month to $335 a month. What percent is the increase?

Keystrokes	Display	
285 [ENTER↑]		
335 [f][Δ%]	*17.54*	% increase.

Example: You forgot to place a stop order and your stock fell from $57.50 to $13.25 a share. What percent is the decrease?

Keystrokes	Display	
57.50 [ENTER↑]		
13.25 [f][Δ%]	*-76.96*	% decrease.

Retail calculations for markup and margin are simple percentage problems using the [Δ%] function. For example, to calculate the markup of an item based on the wholesale or original purchase cost, enter the purchase cost, then key in the selling price and press [f][Δ%]. To

calculate the margin based on the selling price of an item, enter that price, then key in the purchase cost and press [f][Δ%]. Actually, all percent differences follow the same rule: *To find percent increase or percent decrease, enter the old amount (base), then key in the new amount and press* [f][Δ%]. The first number (base) is retained in the automatic memory stack.

Percent of Total [%T]

To find what percentage one number is of a sum of numbers, first calculate the sum and then key in the particular number you wish to convert to a percentage and press [f][%T]

1. Key in the first number.
2. Press [ENTER↑].
3. Key in subsequent numbers (if any), each followed by [+].
4. Key in the particular number you wish to convert to a percentage of that total.
5. Press [f][%T].

Example: What percent is $5.50 of the sum of $10.25 + $5.50 + $3.25 + $21.00? And what percent is $21.00 of the same total?

Keystrokes	Display	
10.25 [ENTER↑]	*10.25*	Enter the first number.
5.50 [+] 3.25 [+] 21 [+]	*40.00*	Key in and add the rest of the numbers.
5.50 [f][%T]	*13.75*	% of total.

The total is retained in the automatic memory stack—simply clear the display, key in a new number and find [%T].

Keystrokes	Display	
[CLx]	*0.00*	
21 [f][%T]	*52.50*	% of total.

LAST X

In addition to the automatic memory stack, the HP-38E contains a separate automatic register called the LAST X register. This register preserves the value that was in the display before the performance of a function. To place the contents of the LAST X register into the display again, press [g] [LAST X]

The [LAST X] key makes it easy to recover from keystroke mistakes, such as pressing the wrong function key or keying in the wrong number. This ease of recovery becomes especially important when you are dealing with large numbers, where reentering may be cumbersome.

Example: Divide $55,499,041 by $7,472 after you have mistakenly divided by $1,472.

Keystrokes	Display	
55499041 [ENTER↑]	*55,499,041.00*	
1472 [÷]	*37,703.15*	Oops! You made a mistake.
[g] [LAST X]	*1,472.00*	Retrieves that last entry.
[×]	*55,499,041.00*	You're back at the beginning.
7472 [÷]	*7,427.60*	The correct answer.

The Hidden Function (Mantissa)

When in any display format, you wish to view the contents of the true mantissa (all 10 digits) press [f] CLEAR [PREFIX] and hold down the [PREFIX] key. All 10 digits of the mantissa held internally will be displayed. Release the key and the display will revert back to its original contents. For more information on display formatting, refer to The Display and Memory in *Your HP Financial Calculator*.

With the data intact from the last example and 7,427.60 still in the display, use the hidden mantissa function to see the rest of the number (hidden digits) held internally.

Keystrokes	Display	
[f] CLEAR [PREFIX]	*7427601847*	Mantissa.

Even though the display returns to 7,427.60, the calculator computes using all 10 digits of the number 7,427.601847.

Keying In Exponents of 10

If you are dealing with millions, billions or trillions of dollars, you can save keystrokes (all those zeros!) by using the [EEX] *(enter exponent of 10)* key. For example, to key in 15.6 trillion (15.6×10^{12}) and multiply it by 25:

Keystrokes	Display		
[CLx]	*0.00*		
15.6	*15.6*		
[9] [EEX]	*15.6*	*00*	
12	*15.6*	*12*	This means 15.6×10^{12}.
[ENTER↑]	*1.560000*	*13*	Displayed in scientific notation.
25 [×]	*3.900000*	*14*	The answer.

This is the same as 390 trillion. (Move the decimal point 14 places to the right.)

Example: The estimated world aggregate of gross national product in 1971 was $3,500,000,000,000. If the 1971 GNP of the United States was $1,151,800,000,000, what was the GNP for the rest of the world?

Keystrokes	Display		
3.5 [9] [EEX]	*3.5*	*00*	
12	*3.5*	*12*	
[ENTER↑]	*3.500000*	*12*	(3.5×10^{12}).
1.1518 [9] [EEX]	*1.1518*	*00*	
12	*1.1518*	*12*	(1.1518×10^{12}).
[−]	*2.348200*	*12*	

The GNP for the rest of the world was $2,348,200,000,000.

(Refer to *Your HP Financial Calculator,* The Display and Memory, for more information on scientific notation display.)

Calendar

In many financial calculations there is a frequent need to find the number of days between two calendar dates, or the calendar date of a certain number of days in the future (or past). These calculations can be accomplished on the HP-38E utilizing the [f] [DATE] and [f] [△DAYS] keys. The day count will be calculated on both a 30-day month, 360-day year (Y-register) and an actual day year including leap days (displayed X-register). The HP-38E can calculate actual days for nearly 2,500 years from October 15, 1582, through November 25, 4046.*

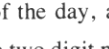
Date Format

In order to use the calendar functions, dates must be entered in the format designated by the payment switch setting.

When the date format D.MY BEGIN M.DY END is set to D.MY (day, month, year):

1. Key in the number of the day, and press the decimal point.
2. Key in the month as a two digit number followed by the full year.

Example: January 3, 2001, is entered as 3.012001.

When the date format D.MY BEGIN  M.DY END is set to M.DY (month, day, year):

1. Key in the number of the month, and press the decimal point.
2. Key in the date as a two-digit number followed by the full year.

Example: July 25, 1978, is entered as 7.251978.

Days Between Dates [△DAYS]

To calculate the number of days between two dates, use the following procedure:

1. Specify the date format.
2. Key in the earlier date and press [ENTER↑].
3. Key in the later date and press [f] [△DAYS].

*Note that early day-of-week displays may be different than those normally reported in history. The reason for this is that prior to September 14, 1752 (the day England and its colonies switched from the Julian Calendar to the current Gregorian Calendar), the Julian Calendar was standard. The Julian Calendar was falling behind by 3 days every 400 years. This problem is corrected in the Gregorian Calendar by eliminating 3 leap years every 400 years. In the Gregorian system, century years (1700, 1800, 1900, etc.) are not leap years unless they are divisable by 400 (2000, 2400, 2800, etc.) The Gregorian Calendar was adopted by other countries at various other dates.

The following values are returned in the stack.

Y	Days calculated on a 30-day month, 360-day year.
X	Actual number of days between dates.

The contents of the X-register are always visible in the display. To view the contents of the Y-register, use [x≷y] to exchange the contents of the X- and Y-registers. (The automatic memory stack and methods for manipulation of the register contents are covered in detail in *Your HP Financial Calculator,* appendix A.)

Example: How many actual days are there between September 1, 1977, and January 3, 2001? How many days are there between the two dates based on a 30-day month, 360-day year?

Set the date format D.MY BEGIN M.DY END to M.DY:

Keystrokes	Display	
9.011977 [ENTER↑]	*9.01*	
1.032001 [f] [△DAYS]	*8,525.00*	Actual number of days.
[x≷y]	*8,402.00*	Number of days based on 30-day month.

Future or Past Dates [DATE]

Determining a future (or past) date that is a fixed number of days before or after a given date can be accomplished using [f] [DATE] and the following keystrokes:

1. Specify the date format.
2. Key in the date and press [ENTER↑].
3. Key in the number of days, and press [f] [DATE]. Use a negative number to find a day in the past.

Example: If you purchased a 120-day option on a piece of land on May 16, 1978, what would the expiration date be? Let's enter the day first. Set the date format D.MY BEGIN M.DY END to D.MY.

Keystrokes	Display	
16.051978 ENTER↑	*16.05*	
120 f DATE	*13,09,1978 3*	Wednesday, September 13, 1978.

The number on the far right of the display designates the day of the week.

1 is Monday.
2 is Tuesday.
3 is Wednesday.
4 is Thursday.
5 is Friday.
6 is Saturday.
7 is Sunday.

When f DATE is executed as an instruction in a running program, the calculator automatically pauses for about 1 second to display the result, then resumes program execution.

Section 2

The Financial Functions

Your calculator has the most frequently used business calculations preprogrammed into the top row of keys. Note that some complex financial functions cause the display to blink on and off for several seconds, indicating that the calculation is being performed.

[n]	Number of periods.[1]
[i]	Interest rate per period.[1]
[PV]	Present value.[1]
[PMT]	Payment.[1]
[FV]	Future value.[1]
[12×]	Multiply by 12, store in n.
[12÷]	Divide by 12, store in i.
[AMORT]	Amortization.[2]
[INT]	Simple interest.[2]
[NPV]	Net present value.[2]
[IRR]	Internal rate of return.[2]
[CF_0]	Initial cash flow.[3]
[CF_j]	Subsequent cash flows.[3]
[N_j]	Number of consecutive equal cash flows.[3]

The Financial Registers

Special memories, called the financial storage registers, are reserved for financial interest calculations. To enter data into a financial register, simply key in the number and press the financial key. The financial keys either store or solve (see above).

---

[1] Solve and store function (has corresponding financial register).

[2] Solve-only function (no corresponding financial register).

[3] Store-only function. Discounted cash flow analysis utilizes up to 20 of the storage registers, R_0 through R_9 and $R_{.0}$ through $R_{.9}$. N_j has 21 separate memories for storing the number of times (0 through 99) a cash flow occurs. The N_j memories automatically assume 1, unless otherwise specified.

Displaying Financial Values

Any of the values associated with the financial registers can be recalled by pressing [RCL] followed by the appropriate key (e.g., [RCL] [PV]).

Clearing the Financial Registers

Each time you begin a new problem press [f] CLEAR [FIN] to erase previous financial values. When you press [f] CLEAR [FIN], the previous financial register values are replaced with zeros. The display remains unchanged. If you want to change some, but not all, of the values in a financial problem, it is not necessary to press [f] CLEAR [FIN] and reenter all of the values again. Simply key in the new data and press the appropriate financial keys to change particular financial values.

Reentering Values

Once a value is stored in a particular register, it remains in the register for possible future use until it is either overwritten (replaced by another number) or the calculator is switched off.

There are four ways to overwrite (and consequently change) values stored in the financial registers.

1. Pressing [f] CLEAR [ALL] or [f] CLEAR [FIN].
2. Keying in a different number and pressing the original key. This stores the new value in the place of the original value.
3. Using the [AMORT] function. This function changes both the [n] and [PV] values.
4. Using the discounted cash flow analysis functions [CF_0] and [CF_j] also change the value in [n].

The Payment Switch

The payment switch D.MY BEGIN M.DY END serves two independent purposes. When doing compound interest calculations it is used to differentiate between payments made at the beginning of each period (BEGIN) or at the end of each period (END). For a more thorough explanation, refer to *Your HP Financial Calculator*.

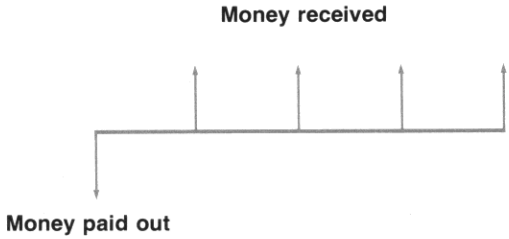
As you already know, this switch also designates the date format used with the calendar functions. (Refer to page 24.)

Financial Interest Calculations

The applications and examples here are representative of a wide range of possible calculations. If your specific problem is not included in the pages that follow, *don't* assume that your calculator won't solve it. Many problems in finance involve terminology that differs from industry to industry. The basic concepts are the same, but we all speak our own language, the vernacular of our profession.

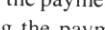
The financial symbols n, i, PMT, PV, and FV, along with the concepts of simple and compound interest, amortization, and discounted cash flow analysis are explained in *Your HP Financial Calculator: An Introduction to Financial Concepts and Problem Solving*. We strongly encourage you to read the financial section of the book as it includes a discussion on how to solve virtually any compound interest problem with your calculator by using a cash flow diagram. A cash flow diagram enables you to describe a compound interest problem in terms that the calculator can understand. Once you draw and label your diagram, you simply key in the known data and solve for an unknown value.

If the solution to your problem isn't evident at first, construct a cash flow diagram—a picture of money received and money paid out.



Once you've done this, label your diagram with all of the known data that pertains to the problem: interest rate, duration of the transaction, number of compounding periods, payment amounts, amount of the loan or investment, etc. Instead of "What is my problem?" ask yourself, "What are the cash flows?"

Solving for any of the top row values (n, i, PV, PMT, or FV) is easy with your calculator. There are four simple rules to remember—rules that are the same for all compound interest calculations:

1. Given three or four of the financial values (n, i, PV, PMT, or FV), you can solve for the fourth and/or fifth values, as long as n and/or i are known.* Both n and i are involved in all financial calculations. You can enter the values in any order.
2. Use the cash flow sign convention throughout all compound interest calculations (including amortization): *Cash received (arrow pointing up) is represented by a positive value (+). Cash paid out (arrow pointing down) is represented by a negative value (−).*
3. Whenever payments (PMT) are involved, it is always necessary to specify whether the payments are made at the beginning of the payment period or whether the payments are made at the end of the payment period, by setting the payment switch D.MY BEGIN  M.DY END to the proper position.
4. Remember that [n] and [i] must correspond to the same time frame. If n is months, then i must be the monthly interest rate; if n is the number of quarterly compounding periods, i must be the quarterly interest rate.

Remember, the n value represents the total number of compounding or payment periods. The alternate function, [g] [12×], converts yearly periods to monthly periods ($12x$) then automatically stores that number in [n]. To enter 30 years, press 30 [n]. If you wish to input that in monthly periods, press 30 [g] [12×]. The calculator converts 30 (years) to 360 (months) and stores it automatically in [n]. There is no need to press [n] again.

The i value is the interest rate per period. If interest is expressed as an annual rate compounding monthly, pressing [g] [12÷] calculates the interest rate per month and automatically stores it in [i]. To enter 9% annual interest press 9 [i]. To input the monthly rate, press 9 [g] [12÷]. There is no need to press [i] again, the monthly rate is automatically stored.

* The calculator uses all four variables to solve for the fifth. Zero is assigned to those values that have not been computed or entered since the calculator was last cleared.

The [PMT] key stands for periodic payment* or deposit amount. It assumes equal periodic payments and must correspond to the same time frame as [n] or [i].

The [PV] key stands for present value;* the amount of money at the start of a transaction or the discounted amount of a future cash flow.

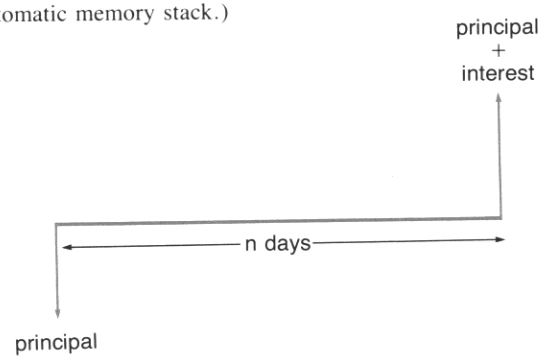
The [FV] key represents the future value* of money or the amount you will obtain/pay at the end of the term. Or you can use [FV] to solve for a balloon payment at the end of a transaction.

The following examples show some of the solutions for each of the financial values. If your problem does not match the example, draw a cash flow diagram, key in what you *do* know, using the conventions outlined above, and solve for the unknown value. You'll find that you can consider numerous investment alternatives with just a few easy keystrokes.

Simple Interest [INT]

Accrued Interest—360-Day Basis

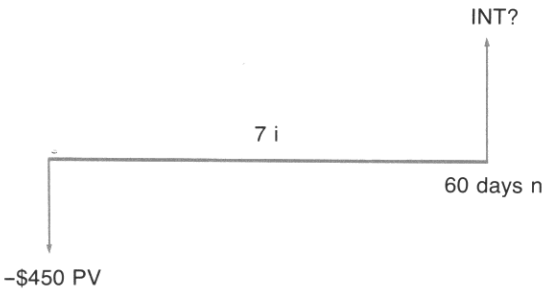
This calculation finds the amount of accrued simple interest on a 360-day basis when the number of days, interest rate, and principal (present value) are known. This calculation returns the principal to the Y-register, after changing its sign, so that it can be added to the interest calculated. (Refer to *Your HP Financial Calculator* for a description of the automatic memory stack.)



* Remember the cash flow sign convention!

1. Input the following in any order:
   - Key in the number of days, press [n].
   - Key in annual interest rate, press [i].
   - Key in principal, press [PV].*
2. Press [f] [INT] to obtain the amount of accrued interest on a 360-day basis.
3. Press [+] to obtain the net amount (principal + interest) on a 360-day basis.

Example: Your good friend needs a loan to start his latest enterprise. He has requested that you lend him $450 for 60 days. You lend him the money at 7% simple interest, based on a 360-day year. What is the amount of accrued interest he will owe you in 60 days? What is the net amount?



Keystrokes	Display	
60 [n]	*60.00*	
7 [i]	*7.00*	
450 [CHS] [PV]	*–450.00*	
[f] [INT]	*5.25*	Accrued interest.
[+]	*455.25*	Net amount paid back (received).

* Use the cash flow sign convention!

Accrued Interest—365-Day Basis

This calculation finds the amount of accrued simple interest on a 365-day basis when the number of days, interest rate, and principal (present value) are known. The accrued interest on an actual day basis is returned to the Z-register.

1. Input the following in any order:
   - Key in number of days, press [n].
   - Key in annual interest rate, press [i].
   - Key in principal, press [PV].*
2. Press [f] [INT] [g] [R↓] [x≷y] to obtain the accrued interest on a 365-day basis.
3. Press [+] to obtain the net amount (principal + interest) on a 365-day basis.

Example: What is the accrued interest and net amount on $450 for 60 days at 7%, figured on a 365-day year? (If you left these values in the calculator from the example above, follow the keystroke instructions from step 2.)

Keystrokes	Display	
60 [n]	*60.00*	
7 [i]	*7.00*	
450 [CHS] [PV]	*-450.00*	
[f] [INT] [g] [R↓] [x≷y]	*5.18*	Accrued interest.
[+]	*455.18*	Net amount paid back (received).

Compound Interest

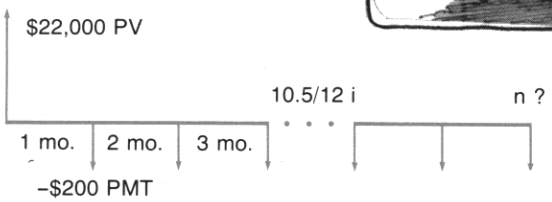
Solving for the Number of Compounding or Payment Periods [n]

1. Set the payment switch, if applicable, to the desired setting and press [f] CLEAR [FIN] which resets [n], [i], [PMT], [PV], and [FV] to zero.

* Use the cash flow sign convention!

2. Input the following in any order:
   - Key in the periodic interest rate, press [i].
   - Key in at least two of the following:
     - present value, press [PV],
     - payment amount, press [PMT], or
     - future value, press [FV].
3. Press [n] to obtain the number of periods.*

Example 1: Fur trapper Bill Buckskin wishes to invest in a $22,000 log cabin to keep the rain off. A local merchant has offered to loan Bill the $22,000 at 10.5% interest. Making $200 monthly payments, how long will it take Bill to repay his mortgage?



Set the payment switch D.MY BEGIN M.DY END to END.

Keystrokes	Display	
[f] CLEAR [FIN]		
10.5 [g] [12÷]	*0.88*	% monthly interest rate.
200 [CHS] [PMT]	*-200.00*	Negative for money paid out.
22000 [PV]	*22,000.00*	Loan amount (received).
[n]	*376.89*	Months.
12 [÷]	*31.41*	Years.

* If n is not an integer and PMT is not zero, then the use of the top row keys produce mathematically correct results with no simple useful interpretation. We show in the following examples, some instances in which the [n] value should be rounded up or down to an integer value, before other values are calculated. Refer to the applications books for bonds and notes applications with non-integer [n] values. If a value of [n] is calculated such that $0.995 \leq n \leq 1$, we assume that it was done unintentionally. Therefore, the [n] value will automatically be rounded to 1, before other top row values are calculated.

We calculated 376.89 payment periods, but chances are, you wouldn't make the last payment separately on a fraction of a month. Let's calculate the fractional payment amount and add it to the regular payment to calculate, in essence, the balloon payment made in the 376th month.

Keystrokes	Display	
376 [n]	*376.00*	Integer portion of number of periods.
[FV]	*-175.88*	Remaining balance after 376 payments.
[RCL] [PMT]	*-200.00*	Regular payment amount.
[+]	*-375.88*	Balloon payment.

What if, instead of a balloon payment, you wanted to make a final short payment? Round the calculated [n] to the next larger integer, then press [FV] to find the amount you should subtract from the payment amount. ([FV] shows the amount that you would have overpaid, had you paid the full payment amount.)

Keystrokes	Display	
377 [n]	*377.00*	[n] rounded up to whole number.
[FV]	*22.58*	Amount to be subtracted from payment.
[RCL] [PMT]	*-200.00*	Payment amount.
[+]	*-177.42*	Final short payment.

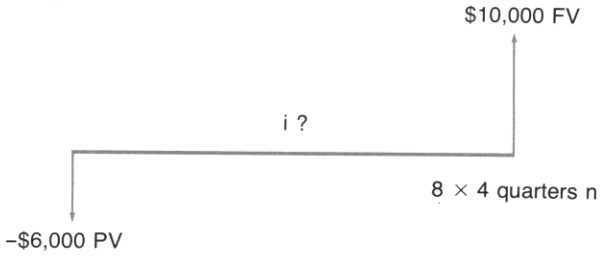
Example 2: A potential oil field site currently appraised at $380,000 appreciates at 30% per year. If this rate continues, how many years will it be before this land is worth $750,000?

Keystrokes	Display	
f CLEAR FIN		Clear financial registers.
30 i	*30.00*	
380000 CHS PV	*-380,000.00*	Remember the sign convention.
750000 FV	*750,000.00*	
n	*2.59*	Years.

Solving for the Periodic Interest Rate i

1. Set the payment switch, if applicable, to the desired setting and press f CLEAR FIN.
2. Input the following in any order:
   - Key in the number of periods, press n.
   - Key in at least two of the following:*
     - payment amount, press PMT,
     - present value, press PV, or
     - future value, press FV.
3. Press i to obtain the periodic interest rate.
4. Key in the number of periods per year, press × to obtain an annual interest rate.

Example 1: What annual interest rate must be obtained to accumulate $10,000 in 8 years on an investment of $6,000, with quarterly compounding?



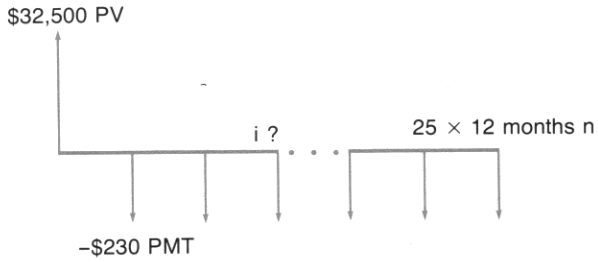
* Remember the sign convention!

Keystrokes	Display	
f CLEAR FIN		
8 ENTER↑ 4 × n	*32.00*	Quarters.
6000 CHS PV	*-6,000.00*	Investment.
10000 FV	*10,000.00*	Total returned.
i	*1.61*	% quarterly interest rate.
4 ×	*6.44*	% annual interest rate.

What if the compounding were monthly?

Keystrokes	Display	
8 g 12×	*96.00*	Months.
i	*0.53*	% monthly interest rate.
12 ×	*6.40*	% annual interest rate.

Example 2: What is the annual interest rate on a 25-year, $32,500 mortgage with $230 monthly payments?



Set the payment switch D.MY BEGIN M.DY END to END.

Keystrokes	Display	
f CLEAR FIN		
25 g 12×	*300.00*	Months.
230 CHS PMT	*-230.00*	Remember the sign convention.
32500 PV	*32,500.00*	Mortgage amount.
i	*0.58*	% monthly interest rate.
12 ×	*7.01*	% annual interest rate.

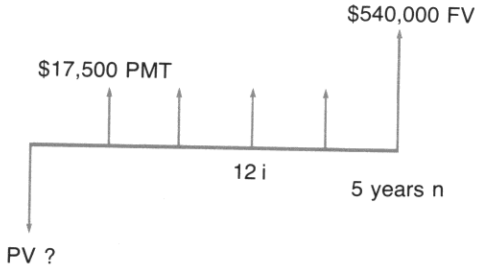
What is the annual interest if there is a balloon payment of $14,000 at the end of the 20th year?

Keystrokes	Display	
14000 CHS FV	*-14,000.00*	Put balloon payment in FV.
20 g 12×	*240.00*	Change the value in n.
i	*0.60*	% monthly interest rate.
12 ×	*7.21*	% annual interest rate.

Solving for Present Value PV

1. Set the payment switch, if applicable, to the desired setting and press f CLEAR FIN.
2. Input the following in any order:
   - Key in the number of periods, press n.
   - Key in the periodic interest rate, press i.
   - Key in the payment amount and press PMT *or* the future value and press FV *or* both, if known.
3. Press PV to obtain the present value.

Example 1: A development company is purchasing a group of condominiums with an annual net cash flow of $17,500. The expected holding period is 5 years with an estimated selling price of $540,000 at that time. If the company wishes to realize a 12% yield, what is the maximum purchase price of the condominiums?

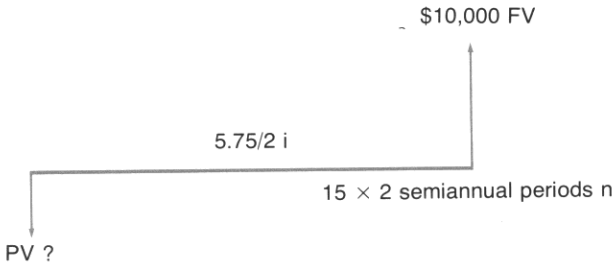


Set the payment switch D.MY BEGIN ▬ M.DY END to END.

Keystrokes	Display	
f CLEAR FIN		
5 n	*5.00*	
12 i	*12.00*	
17500 PMT	*17,500.00*	
540000 FV	*540,000.00*	
PV	*-369,494.09*	Amount that company can pay to realize a 12% yield.

In this case PV represents the maximum price, $369,494.09 necessary to achieve the desired yield.

Example 2: You look forward to retirement in 15 years and wish to deposit one lump sum which will grow to $10,000 at that time, earning 5¾% interest compounded semiannually. How much do you need to deposit today to reach that goal?



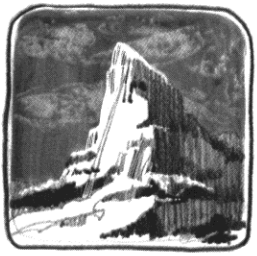
Keystrokes	Display	
f CLEAR FIN		
15 ENTER↑ 2 × n	*30.00*	Semiannual periods.
5.75 ENTER↑ 2 ÷ i	*2.88*	% semiannual interest rate.
10000 FV	*10,000.00*	Amount needed.
PV	*-4,272.72*	Principal to be invested.

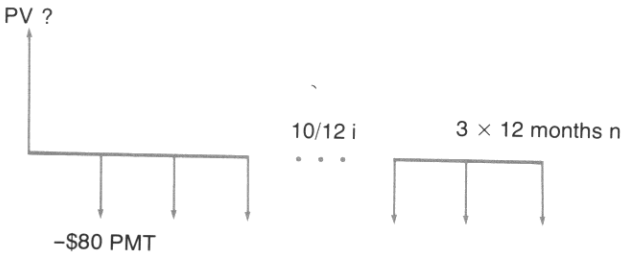
What if you also make semiannual deposits of \$50?

Set the payment switch D.MY BEGIN ▬▬ M.DY END to END.

Keystrokes	Display	
50 [CHS] [PMT]	*-50.00*	Payment amount.
[PV]	*-3,276.67*	Principal to be invested.

Example 3: Rather than buying a dog sled, Yellowstone Sam decides to purchase a snowmobile. If he plans to pay \$80 per month for 3 years, and if he's willing to pay 10% annual interest, how much can he afford to pay for the snowmobile?





Set the payment switch D.MY BEGIN ▬▬ M.DY END to END.

Keystrokes	Display	
[f] CLEAR [FIN]		
3 [g] [12×]	*36.00*	Months.
10 [g] [12÷]	*0.83*	% monthly interest rate.
80 [CHS] [PMT]	*-80.00*	Monthly payment.
[PV]	*2,479.30*	Price he can afford for snowmobile.

Solving for the Periodic Payment Amount [PMT]

1. Set the payment switch to the desired setting and press [f] CLEAR [FIN].
2. Input the following in any order:
   - Key in the number of periods, press [n].
   - Key in the periodic interest rate, press [i].
   - Key in the present value and press [PV] *or* the future value and press [FV] *or* both, if known.
3. Press [PMT] to obtain the payment amount.

Example 1: Leaving the data from our last example in the calculator, what would Yellowstone Sam's monthly payments be if he finds a snowmobile for $2,150 and if the interest rate and duration of the transaction remain the same? Simply change the value in [PV].

Keystrokes	Display	
2150 [PV]	*2,150.00*	
[PMT]	*-69.37*	Monthly payments.

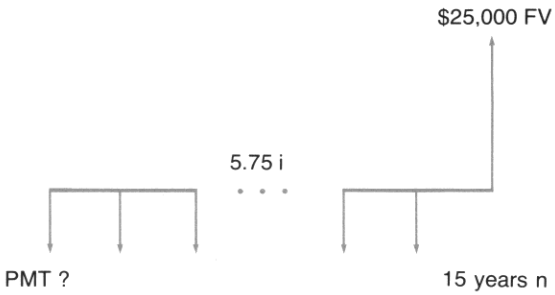
Example 2: Find the monthly payment amount on a 20-year, $27,000 mortgage with an 8.5% annual interest rate.

Set the payment switch

to END.

Keystrokes	Display	
[f] CLEAR [FIN]		
20 [g] [12×]	*240.00*	Months.
8.5 [g] [12÷]	*0.71*	% monthly interest rate.
27000 [PV]	*27,000.00*	Loan amount.
[PMT]	*-234.31*	Monthly payment.

Example 3: Calculate the annual payment amount necessary to accumulate $25,000 in 15 years at 5¾% annual interest.



Set the payment switch D.MY BEGIN M.DY END to BEGIN.

Keystrokes	Display
f CLEAR FIN	
15 n	*15.00*
5.75 i	*5.75*
25000 FV	*25,000.00*
PMT	*-1,035.17*

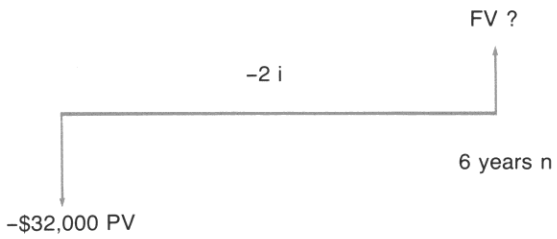
If your deposits are limited to $1,000 per year, how much will have accumulated over the same time period?

Keystrokes	Display
1000 CHS PMT	*-1,000.00*
FV	*24,150.72*

Solving for the Future Value FV

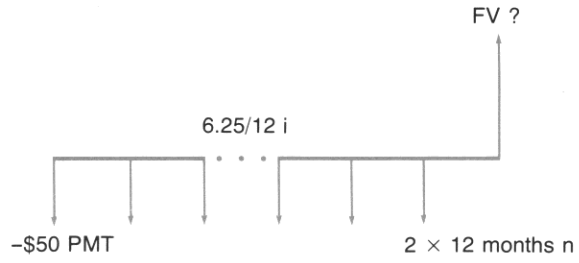
1. Set the payment switch, if applicable, to the desired setting and press f CLEAR FIN
2. Input the following in any order:
   - Key in the number of periods, press n.
   - Key in the periodic interest rate, press i.
   - Key in the payment amount and press PMT *or* the present value and press PV *or* both, if known.
3. Press FV to obtain the future value.

Example 1: Property values in an unattractive area are declining at the rate of 2% per year. What will property presently valued at $32,000 be worth in 6 years if this trend continues?



Keystrokes	Display	
f CLEAR FIN		
6 n	*6.00*	
2 CHS i	*-2.00*	% interest rate.
32000 CHS PV	*-32,000.00*	
FV	*28,346.96*	Property value.

Example 2: If you can afford to deposit $50 per month (beginning today) in an account with 6¼% interest compounded monthly, how much will you have 2 years from now?



Set the payment switch D.MY BEGIN M.DY END to BEGIN.

Keystrokes	Display	
f CLEAR FIN		
2 g 12×	*24.00*	Months.
6.25 g 12÷	*0.52*	% monthly interest rate.
50 CHS PMT	*-50.00*	Monthly deposits.
FV	*1,281.34*	Savings.

If the interest rate remained the same, what monthly deposit amount would be necessary to have a savings of $1,500 in 2 years?

Keystrokes	Display	
1500 FV	*1,500.00*	
PMT	*-58.53*	Monthly deposit.

If you want to leave the deposit amount at $50 per month, how long would it take to accumulate $1,500?

Keystrokes	Display	
50 CHS PMT	*-50.00*	
n	*27.81*	Months.
12 ÷	*2.32*	Years.

Amortization AMORT

You can use your calculator to compute the accumulated interest, principal portion of your payments, and remaining balance of your loan, at any point in time.

1. Press f CLEAR FIN.
2. Set the payment switch D.MY BEGIN M.DY END to the desired setting.

   Note: Use the cash flow sign convention; positive values for cash received and negative values for cash paid out.

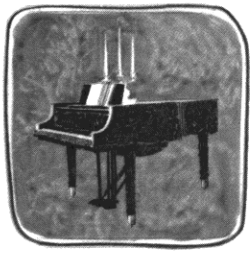
3. Input the following in any order:
   - Key in the principal (amount of loan), press PV.
   - Key in the periodic interest rate, press i.
   - Key in the payment amount, press PMT.

4. Key in the number of periods to be amortized, press [f] [AMORT] to find the accumulated interest on the loan.
5. Press [x≷y] to obtain the portion of the payments made toward the principal, over the specified period of time.
6. Press [RCL] [PV] to obtain the remaining balance of the loan.
7. Press [RCL] [n] to obtain the total number of periods amortized.

In amortization, all payments to principal and payments to interest are rounded to match the display setting. If you compare your calculated answers to the statements of savings and lending institutions, the amount may differ by a few cents, due to different rounding techniques. The normal display shows numbers as dollars and cents. If your problem requires other rounding, set the display to the number of digits you wish carried. (Refer to *Your HP Financial Calculator*, The Display and Memory.)

Note: The [AMORT] function changes two of the top row values: [n] and [PV]. [PV] brings back the new balance, [n] provides the total number of periods amortized.

Example: Pianist Marcella Musica has finally decided to purchase the $14,000 grand piano of her dreams. If she takes out a loan at 10% interest per annum and wants to pay it off in 15 years, how much will her monthly payment be? After you calculate the monthly payment amount, reset [n] to zero and find the accumulated interest and principal portions of her payments and the remaining balance of her loan after 5 years.



Set the payment switch D.MY BEGIN M.DY END to END.

Keystrokes	Display	
[f] CLEAR [FIN]		Clear financial registers.
14000 [PV]	*14,000.00*	Loan amount.
10 [g] [12÷]	*0.83*	Periodic interest rate.
15 [g] [12×]	*180.00*	Payment periods, 15 years times 12 months per year.
[PMT]	*-150.44*	Monthly payment.

Marcella's payments will be $150.44 each month. Now reset [n] and amortize.

Keystrokes	Display	
0 [n]	*0.00*	Set [n] to zero.
5 [ENTER↑] 12 [×]	*60.00*	Number of periods to be amortized in 5 years.
[f] [AMORT]	*-6,411.13*	Accumulated interest portions of payments made over 5 years.
[x≷y]	*-2,615.27*	Accumulated principal portions of payments made over 5 years.
[RCL] [PV]	*11,384.73*	Remaining balance.
[RCL] [n]	*60.00*	Number of periods amortized.

What portion of the payments made in the next 5 years or 60 compounding periods (years 6 through 10) pay off the accumulated interest, and what portion of these payments pay part of the principal? What is the remaining balance after 10 years?

Keystrokes	Display	
60 [f] [AMORT]	*-4,723.41*	Interest portion of payments made in year 6 through year 10.
[x≷y]	*-4,302.99*	Principal portion of payments made in year 6 through year 10.
[RCL] [PV]	*7,081.74*	Remaining balance.
[RCL] [n]	*120.00*	Total number of periods amortized.

You can even generate a complete amortization schedule, one payment period at a time, by simply pressing 1 [f] [AMORT] for each period. If you start at the beginning of the schedule, be sure to change [PV] back to the original principal (as [PV] has been keeping a record of the remaining balance) and change [n] to zero. Generate the first 2 months of the schedule:

Keystrokes	Display	
14000 PV 0 n	*0.00*	
1 f AMORT	*-116.67*	Interest portion of first payment.
x≷y	*-33.77*	Principal portion of first payment.
RCL PV	*13,966.23*	Remaining balance of loan.
1 f AMORT	*-116.39*	Interest portion of second payment.
x≷y	*-34.05*	Principal portion of second payment.
RCL PV	*13,932.18*	Remaining balance of loan.

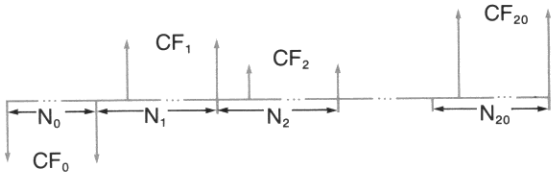
If you wish to calculate the remaining balance alone, simply use the FV key. Let's calculate the remaining balance of Marcella's loan, again after 10 years, to compare with our previous answer.

Keystrokes	Display	
14000 PV	*14,000.00*	Original principal.
120 n	*120.00*	10 years times 12 months.
FV CHS	*7,081.70*	Remaining balance.

Notice that the remaining balance calculated using FV is 4 cents less than the remaining balance calculated using AMORT. That's because AMORT causes each individual payment to be rounded, while FV does not cause numbers to be rounded internally.

Discounted Cash Flow Analysis

Two forms of discounted cash flow analysis are the net present value (NPV) approach and the internal rate of return (IRR) approach. Both of these functions, NPV and IRR, are preprogrammed in your calculator for up to 20 groups of cash flows (excluding the initial investment) with each group containing a maximum of 99 equal cash flows.



When solving for NPV or IRR, use the [CF0], [CFj], and [Nj] keys to store the cash flows as follows:

Use [CF0] *(cash flow zero)* to store the amount of the initial investment. ([CF0] can also be grouped as "a set of equal payments constituting an initial investment.") This key stores the investment amount in storage register R_0 and automatically sets the [n] register to zero. The [n] register serves as an index to the number of distinct cash flows entered for your convenience in editing and correcting entries. (Refer to page 56.)

The positive or negative amounts of future cash flows are stored with [CFj]. The [CFj] key stores the cash flows in the available storage registers (R_1 through R_9, $R.0$ through $R.9$) with the first cash flow in R_1, the second cash flow in R_2, and so on, up to the 20th cash flow, which is automatically stored in [FV]. Each time you press [CFj], the contents of the [n] register are incremented by 1, to count the number of distinct cash flows entered.

The [Nj] key is designed to accomodate those situations where a series of cash flows are equal. Unless otherwise specified, the calculator automatically assumes that N_j is 1. If the cash flow occurs only once, there is no need to enter 1 for N_j.

Net Present Value [NPV]

Assuming a given interest rate, the net present value method sums the present values of the future cash flows and adds it to an initial cash flow. With the HP-38E you can solve directly for the net present value of up to 20 groups of cash flows (excluding the initial investment). The following keystrokes are used to find the net present value of an investment. The position of the payment switch is irrelevant to this calculation.

1. Press [f] CLEAR [ALL].
2. Key in the given interest rate, and press [i].
3. Key in the initial investment amount, and press [g] [CF0]. If there is no initial investment, key in 0 for the amount.

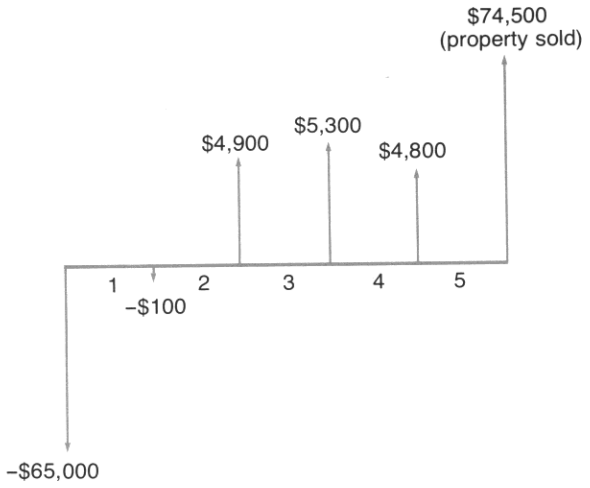
Note: Use the cash flow sign convention; positive values for cash received and negative values for cash paid out.

4. Key in the first cash flow amount and press [g] [CFj], then key in the number of times that cash flow occurs, if other than 1, and press [g] [Nj]. Key in the remaining cash flows in the same manner: first key in the cash flow amount and press [g] [CFj], then key in the number of times that cash flow occurs, if other than 1, and press [g] [Nj]. The 20th cash flow is automatically stored in [FV].*
5. Press [f] [NPV] to display the net present value. The calculated value for NPV is automatically stored in the [PV] register.

Let's take two examples, one using all distinct cash flows and one where we can use [Nj]. If you do not derive the correct answers, refer to page 56 to review and edit your cash flow entries.

Example 1: An investor pays $65,000 for a duplex that he intends to keep 5 years and then sell. The first year he knows he will have to spend a considerable amount for repairs. Will he achieve a desired 9% after-tax yield with the following after-tax cash flows? Note that the duplex is sold for $74,500 at the end of the fifth year.

* If [g] [MEM] displays *P-15 r-19* when pressed, then the 19th cash flow is automatically stored in [FV]; if [MEM] shows *P-22 r-18*, then the 18th cash flow is stored in [FV], and so on. Refer to Automatic Memory Allocation (page 86) for a discussion of the [MEM] function.



Keystrokes	Display	
f CLEAR ALL	*0.00*	Clear financial and storage registers.
9 i	*9.00*	Desired yield.
65000 CHS g CF_0	*−65,000.00*	Initial investment.
100 CHS g CF_j	*−100.00*	First cash flow.
4900 g CF_j	*4,900.00*	Second cash flow.
5300 g CF_j	*5,300.00*	Third cash flow.
4800 g CF_j	*4,800.00*	Fourth cash flow.
74500 g CF_j	*74,500.00*	Fifth cash flow.
f NPV	*−5,054.61*	Net present value.

Since NPV is negative, the investment does not achieve the desired 9% yield. What is the actual internal rate of return? Press f IRR. You'll find that IRR is 7.10, i.e., the yield or return on the investment is 7.1%. Solve again for NPV and you'll find that your answer will be very close to zero, indicating that 7.1% is the actual yield.

For future reference in editing your cash flows press RCL n. The display will show *5.00* indicating that you have stored 5 distinct cash flows (excluding the initial investment).

Example 2: An investor has an opportunity to purchase a piece of property for $70,000. If the going rate of return on this type of investment is 13.75%, and the after-tax cash flows are forecast as follows, should the investor purchase the property?

Year	Cash Flow ($)
1	$14,000
2	11,000
3	10,000
4	10,000
5	10,000
6	9,100
7	9,000
8	9,000
9	4,500
10	71,000

← (property sold in 10th year)

Keystrokes	Display	
f CLEAR ALL	*0.00*	
13.75 i	*13.75*	Interest rate.
70000 CHS g CF_0	*-70,000.00*	Initial investment.
14000 g CF_j	*14,000.00*	First cash flow amount.
11000 g CF_j	*11,000.00*	Second cash flow amount.
10000 g CF_j	*10,000.00*	Third cash flow amount.
3 g N_j	*3.00*	Number of times that cash flow occurs.
9100 g CF_j	*9,100.00*	Fourth cash flow amount.
9000 g CF_j	*9,000.00*	Fifth cash flow amount.
2 g N_j	*2.00*	Number of times that cash flow occurs.
4500 g CF_j	*4,500.00*	Sixth cash flow amount.
71000 g CF_j	*71,000.00*	Seventh cash flow amount.
RCL n	*7.00*	Verifies that you have entered seven distinct cash flows.
f NPV	*879.93*	Net present value.

Since the net present value is positive, the investment meets the profit objectives of at least 13.75% minimum yield. Notice that we stored *distinct* cash flow amounts with [CFj], and instead of keying in each cash flow, we keyed in the number of times that the cash flow occurred (if other than once) and pressed [Nj].

What is the actual yield on this investment? Press [f] [IRR]. (You should get 14.01.)

Internal Rate of Return [IRR]

Internal rate of return (IRR) is an interest rate that equates the present value of a set of cash flows with an initial investment. It is the interest rate that is obtained when the calculated net present value of a series of cash flows is zero. IRR is also called the **yield** or **discounted rate of return.**

On your HP-38E, the IRR function solves for the internal rate of return of up to 20 groups of uneven cash flows (excluding the initial investment), with each group containing a maximum of 99 cash flows. The cash flow sequence must contain at least *one* sign change. In other words, if the first several cash flows are negative, and they are followed by all positive values, that counts as one sign change. Cash flows with multiple sign changes may have multiple answers.

Solutions to IRR will fall into one of the following categories. For a more detailed description and interpretation of the IRR computation refer to appendix B.

Case 1. You get an answer. If a number is displayed, then for the vast majority of IRR problems, this IRR solution is unique.

Case 2. *Error 7* **.** The problem has no solution to IRR. Review your cash flow entries to look for mistakes in data entry. Does the sequence of cash flows show *at least one* sign change?

Case 3. *Error 3* **.** The calculator does not have enough information to continue the IRR computation. There may be multiple solutions to IRR in the mathematical sense or the calculator may not be able to recognize the problem as defined by the sequence of cash flows. If you wish to continue searching for an IRR solution(s), you may do so using a procedure described in appendix B.

The IRR computation is extremely complex, thus your calculator's display may blink on and off for several seconds or even minutes depending on the complexity of the calculation. When the calculator solves for [i] or [IRR], an iteration is used. This means that the calculator systematically obtains a sequence of trial values and, in a series of successive approximations, improves each trial value on the basis of the previous value until an answer is reached. If you think that the calculator is taking an unusual amount of time to find the answer, pressing any key will stop the IRR iterative process.* The display will show the interest rate that the calculator has computed up to this point in the iteration. You can check this number using the net present value method. With the solution for IRR as the interest rate, solve for NPV. The net present value should be close to zero. (Due to rounding, the answer may not be exactly zero.) If you wish to continue the iterative procedure, refer to appendix B.

The keystroke procedure for solving [IRR] is as follows:

1. Press [f] CLEAR [ALL].
2. Key in the amount of the initial investment and press [g] [CF_0]. If there is no initial investment, key in zero for the amount. (Remember, [CF_0] can also be grouped as "a set of equal payments constituting an initial investment," in which case you enter the payment amount, press [g] [CF_0], then enter the number of payments and press [g] [N_j].)

   Note: Use the cash flow sign convention; positive values for cash received and negative values for cash paid out.
3. Key in the amount of the first cash flow and press [g] [CF_j]. Then key in the number of times that cash flow occurs, if other than 1, and press [g] [N_j]. Enter the remaining cash flows in the same manner; first key in the cash flow amount and press [g] [CF_j], then key in the number of times that cash flow occurs and press [g] [N_j]. The 20^{th} cash flow will be stored in the [FV] register automatically.†
4. Press [f] [IRR] to calculate and display the periodic internal rate of return. The calculated value will automatically be stored in [i].

* Providing that at least one iteration has been completed. The HP-38C halts the iterative process immediately. The HP-38E halts when the current iteration has been completed.

† See footnote, page 50.

Zero should be entered for periods with no cash flow.

The calculated answer is the periodic rate of return. If the cash flow periods are other than annual (monthly, quarterly) the answer should be multiplied by the number of periods per year to determine the annual internal rate of return.

Example: An income property is available for $50,000. The annual income over a 23-year projection period (all payments received at the end of the year) may be grouped as follows:

Cash Flow (CF_j)	Number of Years (N_j)
$9,000	First 5 years
$7,500	Next 4 years
$6,000	Next 4 years
$7,500	Next 3 years
$5,000	Last 7 years

What is the annual rate of return?

Keystrokes	Display	
f CLEAR ALL	*0.00*	
50000 CHS g CF_0	*-50,000.00*	Initial investment.
9000 g CF_j	*9,000.00*	First cash flow amount.
5 g N_j	*5.00*	Five years (number of times cash flow occurs).
7500 g CF_j	*7,500.00*	Second cash flow amount.*
4 g N_j	*4.00*	Four years.
6000 g CF_j	*6,000.00*	Third cash flow amount.
4 g N_j	*4.00*	Four years.
7500 g CF_j	*7,500.00*	Fourth cash flow amount.*
3 g N_j	*3.00*	Three years.
5000 g CF_j	*5,000.00*	Fifth cash flow amount.
7 g N_j	*7.00*	Seven years.
RCL n	*5.00*	Verify number of cash flows entered.
f IRR	*15.27*	Annual IRR of 15.27%.

Check your answer with NPV .

* Note that $7,500 occurs two different places in the time scheme. Since the timing of cash flows is important for an IRR calculation, we can only lump together *consecutive* equal payments.

Keystrokes	Display
f NPV	*0.00001*

Since NPV is close to zero, the calculated interest rate is correct.

Changing Cash Flow Entries

1. To change a CF_j value, enter the new CF_j value and press STO j, where j is 0 through 9 or .0 through .9. Since the cash flow amounts are held in the storage registers, you simply change the storage register contents.
2. To change an N_j value, set the n register to indicate which N_j entry you wish to change (0 through 20). Then key in the new N_j value and press N_j.
3. Remember to reset the n register to the number of CF_j entries after you change or review the cash flows.

Example: With the data intact from our last example, change the third cash flow entry (6,000) to 7,000 and change the corresponding N_j value (4) to 5. Then find the new IRR.

Keystrokes	Display	
7000 STO 3	*7,000.00*	CF_3 stored in R_3.
3 n	*3.00*	
5 g N_j	*5.00*	N_3 changed to 5.

Before you press NPV or IRR you must reset the n register to the number of CF_j entries.

Keystrokes	Display	
5 n	*5.00*	Enter the number of CF_j entries.
f IRR	*15.69*	New annual IRR of 15.69%.

Reviewing Cash Flow Entries

If you wish to review all of your cash flows before you press NPV or IRR, use the following methods.

1. **Individual cash flow entries.** Press RCL and the address of the storage register containing the cash flow you wish to review, or press RCL g CF_j to review the CF_j entries in the exact opposite order that you entered them.

2. **Grouped cash flow entries.** Press [RCL] [g] [N_j] [RCL] [g] [CF_j] to review cash flows and group numbers in the exact opposite order that you entered them.
3. Remember to reset the [n] register to the number of [CF_j] entries after you change or review the cash flows.

For example, let's review the cash flows of the IRR problem to make sure that the changes have been made in the right places.

Keystrokes	Display	
[RCL] [g] [N_j]	*7.00*	Last [N_j] entry.
[RCL] [g] [CF_j]	*5,000.00*	Last [CF_j] entry.
[RCL] [g] [N_j]	*3.00*	Fourth [N_j] entry.
[RCL] [g] [CF_j]	*7,500.00*	Fourth [CF_j] entry.
[RCL] [g] [N_j]	*5.00*	Third [N_j] entry.
[RCL] [g] [CF_j]	*7,000.00*	Third [CF_j] entry.
[RCL] [g] [N_j]	*4.00*	Second [N_j] entry.
[RCL] [g] [CF_j]	*7,500.00*	Second [CF_j] entry.
[RCL] [g] [N_j]	*5.00*	First [CF_j] entry.
[RCL] [g] [CF_j]	*9,000.00*	First [CF_j] entry.
[RCL] [g] [N_j]	*1.00*	N_j value for initial investment.
[RCL] [g] [CF_j]	*−50,000.00*	Recalls initial investment.

As you can see, it's as if the calculator has a table inside and reads the entries backwards, starting from the bottom.

[n]	[CF_j]	[N_j]	
0	−50,000 (CF_0)	1	
1	9,000	5	
2	7,500	4	
3	7,000	5	
4	7,500	3	
5	5,000	7	← Reviewing procedure starts here.

Remember to reset [n] to the number of CF_j entries (excluding CF_0) after you review the cash flows.

Section 3:

Mathematical Functions

Reciprocals [1/x]

When you multiply the reciprocal of a number times the number itself, you get one. For instance, the reciprocal of 2 is ½ or 0.5. Two times ½ equals one. To calculate the reciprocal of a number in the display, key in the number, then press [f] [1/x]. For this example, let's change the display to show all nine decimal places by pressing [f] 9.

Example: Add the reciprocal of 9 to the reciprocal of 11; $^1/_9 + {}^1/_{11} = ?$

Keystrokes	Display	
[f] 9	*0.000000000*	Assumes no results remain from previous examples.
9 [f] [1/x]	*0.111111111*	Reciprocal of 9.
11 [f] [1/x]	*0.090909091*	Reciprocal of 11.
[+]	*0.202020202*	Sum of reciprocals.

Now find the reciprocal of the number in the display and then change the display back to show only two decimal places: now you are finding $\frac{1}{1/9 + 1/11}$.

Keystrokes	Display	
[f] [1/x]	*4.950000000*	Reciprocal of sum of reciprocals.
[f] 2	*4.95*	Back to original display format.

Square Root [√x]

To calculate the square root of a number, key in the number, then press [f] [√x].

Solve	Keystrokes	Display
$\sqrt{25}$	25 f $\sqrt{x}$	*5.00*
$\sqrt{81}$	81 f $\sqrt{x}$	*9.00*

You must use a positive number. You cannot calculate the square root of a negative number; that's an illegal operation.

You can also extract higher roots, like cube roots and fourth roots by using the y^x key, not the $\sqrt{x}$ key.

Exponentiation: Raising a Number to a Power y^x

The y^x key raises a positive number to a positive or negative power or a negative number to an integer power. You use it the same simple way you've performed arithmetic operations; the function is executed immediately when you press the key.

1. Key in the base number. This number is designated as y.
2. Press ENTER↑ to separate the first number from the second.
3. Key in the second number (power). This number is designated as x.
4. Press g y^x

In our last example we calculated $\sqrt{81} = 9$. Since we already have 9 in the display, simply press 2 g y^x to get back to 81.

Calculate 3^6:

Keystrokes	Display
3 ENTER↑ 6	*6.*
g y^x	*729.00*

To raise a number to a negative power, follow the same procedure but press CHS to change the sign of your exponent before you perform the operation.

Solve $4.37^{-2.5}$:

Keystrokes	Display
4.37 ENTER↑	*4.37*
2.5 CHS g y^x	*0.03*

To raise a negative number to a positive or negative integer power, key in the base number, press [CHS], press [ENTER↑], key in the integer power, and press [g] [y^x].

Solve $(-2)^3$:

Keystrokes	Display
2 [CHS] [ENTER↑]	*-2.00*
3 [g] [y^x]	*-8.00*

With a negative base, if the exponent is an odd number, the answer will be negative. If the exponent is an even number, your answer will be positive:

$$(-2)^2 = -2 \times -2 = 4$$

$$(-2)^3 = -2 \times -2 \times -2 = -8$$

You can also use [y^x] to raise 0 to a positive power; but of course, your answer will always be zero.

The cube root of a number is that number raised to the ⅓ power. Thus, $\sqrt[3]{n}$ is the same as $n^{1/3}$; the fourth root can be written as $n^{1/4}$ or $n^{0.25}$, etc.

Use the same keystroke sequence that you learned for exponentiation to extract higher roots:

1. Key in the base number and press [ENTER↑].
2. Key in the root desired, then press [f] [1/x].
3. Press [g] [y^x].

To solve $\sqrt[20]{10}$ ($10^{1/20}$):

Keystrokes	Display
10 [ENTER↑]	*10.00*
20 [f] [1/x] [g] [y^x]	*1.12*

Now, try $\sqrt[15]{22}$ ($22^{1/15}$):

Keystrokes	Display
22 [ENTER↑]	*22.00*
15 [f] [1/x] [g] [y^x]	*1.23*

Logarithms [LN]

To calculate the natural logarithm of a number, simply key in the number and press [g] [LN]. To find the $\log_e$ of 30:

Keystrokes	Display
30 [g] [LN]	*3.40*

Now, press [g] [e^x]. Because e^x is the antilog, you return to the original number, 30. To calculate the common logarithm (base 10) of a number, key in the number, then press [g] [LN] 10 [g] [LN] [÷].

To find $\log_{10}$ 5:

Keystrokes	Display
5 [g] [LN]	*1.61*
10 [g] [LN]	*2.30*
[÷]	*0.70*

Example: Logarithms are used in continuous compound interest formulas. If a savings institution offers a 7.79% effective rate on savings compounded continuously, what nominal rate does this represent?

$$\text{Nominal Rate} = 100 \times \ln\left(\frac{7.79}{100} + 1\right)$$

Keystrokes	Display	
7.79 [ENTER↑]	*7.79*	
100 [÷]	*0.08*	
1 [+] [g] [LN]	*0.08*	
100 [×]	*7.50*	Nominal percentage rate.

Antilogarithms [e^x]

To calculate the antilog of a number, key in the number and press [g] [e^x]. This raises e (2.718...) to the power of the value in the display:

Keystrokes	Display
30 [g] [e^x]	*1.068647 13*

Since you used the natural logarithm to convert continuous effective interest to the nominal rate, it follows that the antilog or e^x is used for the opposite conversion.

Example: To compute the continuous effective rate given the nominal rate, the formula is:

$$\text{Continuous Effective Rate} = \left(e^{\left(\frac{\text{Nominal Rate}}{100}\right)} - 1 \right) \times 100$$

So, if a savings institution quotes a nominal rate of 6%, compounded continuously, what is the effective rate?

Keystrokes	Display	
6 [ENTER↑]	*6.00*	
100 [÷]	*0.06*	
[g] [e^x] 1 [−]	*0.06*	
100 [×]	*6.18*	Continuous effective percentage rate.

Factorials [n!]

The n! *(factorial)* key permits you to handle permutations and combinations with ease. To calculate the factorial of a positive integer in the display, press [g] [n!].

Example: Calculate the number of ways that six people can line up for a photograph. Method: $P_6^6 = 6! = 6 \times 5 \times 4 \times 3 \times 2 \times 1$.

Keystrokes	Display	
6	*6.*	
[g] [n!]	*720.00*	The answer.

The calculator overflows for factorials of numbers greater than 69.

Number Alteration

Rounding a Number [RND]

As you know, when you change the number of digits that are displayed beyond the decimal point, the number maintains its full value to 10 digits multiplied by a two-digit exponent of 10 no matter how many digits you see. When you press the [f] prefix key followed by the [RND] *(round)* key, however, the actual number in the calculator becomes the number that is in the displayed X-register. You can change the number of digits rounded by changing the number of digits displayed. For example, if the calculator is set to display four digits beyond the decimal point, the [RND] function will round the internal number to the displayed four digits. The following example shows how the [RND] function can be useful in financial calculations.

Example: Jesse Jones wants to buy a new $150,000 covered swimming pool for his backyard. He decides to finance the pool for 10 years with a final balloon payment. Jesse knows that Aunt Sue's trust fund will pay him $50,000 in 10 years, and he decides to make this his balloon payment. If Jesse gets 9½% financing, what will his monthly payment be? Round the payment amount using the [f] [RND] keys and recalculate the actual balloon. Set the payment switch D.MY BEGIN M.DY END to END.

Keystrokes	Display	
[f] CLEAR [FIN]		Clear the financial registers.
[f] 9		Set calculator to full display.
10 [g] [12×]	*120.0000000*	Calculate number of payments, store in [n].
9.5 [g] [12÷]	*0.791666667*	Calculate monthly interest rate, store in [i].
50000 [CHS] [FV]	*-50,000.00000*	Enter balloon amount.
150000 [PV]	*150,000.0000*	Enter cost of pool.
[PMT]	*-1,689.808909*	Calculate payment.

Keystrokes	Display	
f 2	*-1,689.81*	Set calculator to display two decimal digits.
f RND	*-1,689.81*	Round payment to two places.
f 9	*-1,689.810000*	Return to full display to see rounded number.
PMT	*-1,689.810000*	Store new payment.

Note that the payment was rounded to two decimal places. Since this changed the actual internal value for the PMT, we need to recalculate the balloon amount.

Keystrokes	Display	
FV	*-49,999.78280*	Calculate new balloon amount.
f 2	*-49,999.78*	Return to two decimal digits.

Notice that the rounding of the payment decreased the balloon amount by $0.22.

In scientific notation display, numbers are rounded internally to nine decimal places and are not affected by RND (refer to *Your HP Financial Calculator*).

Integer Portion of a Number INTGR

To extract and display the integer portion of a number (everything to the left of the decimal point), press f INTGR *(integer)*. For example, to display only the integers of the number 123.456:

Keystrokes	Display	
123.456	*123.456*	
f INTGR	*123.00*	Only the integer portion of the number remains.

When f INTGR is pressed, the fractional portion of the number is lost. The entire number, of course, is preserved in the LAST X register.

Fractional Portion of a Number FRAC

To extract and display the fractional portion of a number (everything to the right of the decimal point), press g FRAC *(fraction)*. For example, to see the fractional portion of 123.456 used above:

Keystrokes	Display	
123.456	*123.456*	
g FRAC	*0.46*	Only the fractional portion of the number is displayed, rounded here to normal fixed 2 display.

When g FRAC is pressed, the integer portion of the number is lost. The entire number, of course, is preserved in the LAST X register.

Section 4

Statistical Functions

Summations

The [Σ+] key calculates several different sums and products of values in the X- and Y-registers. The results are automatically accumulated in storage registers R_1 through R_6. Before you begin accumulating a new series of x and y values, you should first clear these registers by pressing [f] CLEAR [Σ]. Whenever you use [Σ+] for a one-variable (x value only) series of accumulations, you should also clear the Y-register of any unwanted values by pressing 0 [ENTER↑] once, just before keying in the first variable in the series.

How it works. When you key in two numbers and press the [Σ+] key, the calculator performs the following accumulations in the indicated registers:

Register	Operation	Data
Displayed X	n	Number of (x, y) entries.
R_1	n	
R_2	Σx	Summation of x values.
R_3	Σx^2	Summation of x^2 values.
R_4	Σy	Summation of y values.
R_5	Σy^2	Summation of y^2 values.
R_6	Σxy	Summation of products of x and y values.

Keying in one number and pressing the [Σ+] key results in the same accumulations as those shown in the table for two-number accumulations. However, as long as the Y-register was first cleared as described above, the values accumulated for Σy, Σy^2, and Σxy will all be zero.

When you input paired data (x and y), key in the y value first, press [ENTER↑] to separate y from x, then key in the x value. The general rule:

y value [ENTER↑] x value [f] [Σ+]

When you input one-number data (x only), simply key in the x value and press [f] [Σ+]

Immediately, you have a powerful data bank for statistical calculations.

Note: If your data x_i or y_i contains many redundant leading digits, you should refrain from copying them into the calculator. For example, if your x-data is 999999999, 1000000001, 1000000002, you should enter the x-data as −1, 1, 2 and add the redundant digits (1000000000) to any x-related answer produced.

Viewing the Statistical Registers

To see any of the statistical accumulations at any time, simply press [RCL] then the address of the desired register (R_1 through R_6). Remember, when you recall a number from a register, only a copy of the number appears in the display.

Deleting and Correcting Data

If you key in an incorrect entry with [Σ+], you don't have to start over again. If you keyed it in before pressing [f] [Σ+], simply press [CLx] to clear the display, then continue on with the correct value.

If you had already added in the wrong value, simply press [g] [LAST x] to recall the wrong number and press [g] [Σ−], then continue with the correct number.

This applies to two variables, as well as one. Suppose you key in 10 [ENTER↑] 20 [f] [Σ+] and discover that the y-value is wrong. Delete the data pair by pressing [g] [LAST x] [g] [Σ−], then continue with the correct numbers. If the incorrect number is not a member of the last data pair you keyed in, then key in both numbers again and delete them with [Σ−] (e.g., press 10 [ENTER↑] 20 [g] [Σ−]).

Mean [x̄]

Your calculator can quickly calculate the means or arithmetic averages of one or two variables. Whether it's the average of test scores or last month's sales figures, given one or two sets of numbers, your calculator will calculate the mean of those samples.

1. Press [f] CLEAR [Σ] to clear old data from the statistical registers.
2. If you are summing one set of numbers, key in the first number and press [f] [Σ+]; then the second number and press [f] [Σ+] again; the third number, etc. Continue until you have entered all the values.

3. If you are summing two sets of numbers, key in the y-value and press [ENTER↑]; key in the x-value, then press [f][Σ+]. Key in the second y-value, press [ENTER↑], key in the second x-value, and press [f][Σ+]. Continue until you have entered all the values.
4. Press [g][x̄] for the mean of the x-values.
5. Press [x⇄y] for the mean of the other set of values (y).

Example: A survey of seven salespersons in your company reveals that they work the following hours a week and sell the following dollar volumes each month. How many hours does the average salesperson work each week? How much does the average salesperson sell each month?

Salesperson	Hours/Week	Sales/Month
1	32	$17,000
2	40	$25,000
3	45	$26,000
4	40	$20,000
5	38	$21,000
6	50	$28,000
7	35	$15,000

To find the average workweek and sales of this sample:

Keystrokes	Display	
[f] CLEAR [Σ]		Statistical registers cleared.
32 [ENTER↑]	*32.00*	
17000 [f][Σ+]	*1.00*	First entry.
40 [ENTER↑]	*40.00*	
25000 [f][Σ+]	*2.00*	Second entry.
45 [ENTER↑]	*45.00*	
26000 [f][Σ+]	*3.00*	
40 [ENTER↑]	*40.00*	
20000 [f][Σ+]	*4.00*	
38 [ENTER↑]	*38.00*	
21000 [f][Σ+]	*5.00*	

Keystrokes	Display	
50 ENTER↑	*50.00*	
28000 f Σ+	*6.00*	
35 ENTER↑	*35.00*	
15000 f Σ+	*7.00*	Total number of entries in the sample.
g x̄	*21,714.29*	Mean dollar sales per month ($\bar{x}$).
x≷y	*40.00*	Mean workweek in hours ($\bar{y}$).

Standard Deviation s

The s function calculates the standard deviation (a measure of dispersion around the mean) of the accumulated data. With the data intact from the previous example, compute s as follows:

Keystrokes	Display	
g s	*4,820.59*	Dollars (s_x).
x≷y	*6.03*	Hours (s_y).

Your calculator computes standard deviation according to the formulas:

$$s_x = \sqrt{\frac{n\,\Sigma x^2 - (\Sigma x)^2}{n\,(n-1)}} \qquad s_y = \sqrt{\frac{n\,\Sigma y^2 - (\Sigma y)^2}{n\,(n-1)}}$$

The formulas for s_x and s_y give *best estimates* of the population standard deviation based on a sample of the population. Thus, current statistical convention calls them *sample* standard deviations. So we have assumed that the seven salespersons are a sample of the population of *all salespersons*, and our formulas derive best estimates of the population from the sample.

What if the seven salespersons constituted the whole population of salespersons. Then we wouldn't need to *estimate* the *population* standard deviation. We can find the *true population* standard deviation (σ) when the data set equals the total population, using the following keystrokes. (It turns out that if you sum the mean of the population into the set itself and find the new s, computed using the formulas above, that s will be the *population* standard deviation, σ, of the original set.)

Keystrokes	Display	
g $\bar{x}$	*21,714.29*	Mean (dollars).
f $\Sigma+$	*8.00*	Number of entries + 1.
g s	*4,463.00*	σ_x.
x≷y	*5.58*	σ_y.

To continue summing data pairs, press g $\bar{x}$ g $\Sigma-$ before entering more data.

Weighted Mean $\bar{x}_w$

You can also compute the weighted mean of a set of numbers, if you know the corresponding weights of the items in question.

1. Press f CLEAR Σ
2. Key in the value of the item and press ENTER↑, then key in its weight and press f $\Sigma+$ Key in the second item's value, press ENTER↑, key in the second weight, and press f $\Sigma+$. Continue until you have entered all the values of the items and their corresponding weights. The rule for inputting the data is "item ENTER↑ weight f $\Sigma+$ "
3. Press g $\bar{x}_w$ for the weighted mean of the items.

Example: Suppose that you stop during a vacation drive to purchase gasoline at four stations as follows: 15 gallons at 59¢ per gallon, 7 gallons at 63¢ per gallon, 10 gallons at 61.5¢ per gallon, and 17 gallons at 65¢ per gallon. You want to find the average cost per gallon of gasoline purchased. If you purchased the same quantity at each station, you could determine the simple arithmetic average or mean using the $\bar{x}$ key. But since you know the value of the item (gasoline) and its corresponding weight (number of gallons purchased), use the $\bar{x}_w$ key to find the weighted mean:

Keystrokes	Display	
f CLEAR Σ		Statistical registers cleared.
59 ENTER↑		
15 f Σ+	*1.00*	First item and weight.
63 ENTER↑		
7 f Σ+	*2.00*	Second item and weight.
61.5 ENTER↑		
10 f Σ+	*3.00*	Third item and weight.
65 ENTER↑		
17 f Σ+	*4.00*	Fourth item and weight.
g $\bar{x}_w$	*62.16*	Weighted mean cost per gallon in cents.

Example 2: What if you purchase shares of Liquidlight stock at the following prices over the course of a year: 80 at $23 per share, 150 at $32 per share, 100 at $28 per share, and 50 at $40 per share. Since you know the value of the stock purchased and its corresponding weight, you can determine the break-even point, i.e., the weighted mean.

Keystrokes	Display	
f CLEAR Σ		Statistical registers cleared.
23 ENTER↑		
80 f Σ+	*1.00*	
32 ENTER↑		
150 f Σ+	*2.00*	
28 ENTER↑		
100 f Σ+	*3.00*	
40 ENTER↑		
50 f Σ+	*4.00*	
g $\bar{x}_w$	*30.11*	Dollars per share; weighted average.

Note that if all of the weights are the same, the weighted mean is the same as the arithmetic mean.

Linear Regression

When you have a set of data points, it is often useful to find out how closely they are related to each other. If you find that the data points are closely related, they can help you make projections or estimates based on known data. Linear regression is a statistical method for defining a straight line that best fits a set of data points, thus providing a relationship between two variables.

Since two points define a line, at least two data points must be in the calculator before a line can be drawn or fitted to them. After you have accumulated the data points using the [Σ+] key, you can quickly estimate other values. But these estimated values are not very reliable unless the original data can be described or generalized to a certain degree of accuracy.

Correlation Coefficient

It's a good idea to check the "goodness of fit" of the linear function, by calculating the correlation coefficient, before estimating other values. This will tell you how close to a straight line the data points lie.

Example: A commercial land appraiser has examined six vacant lots in the downtown section of a local community, all of which have the same depths but different frontages and values. Based on the following input data, what is the relationship between frontage and lot value? In other words, how well does the following data fit a straight line? Would this be a good sample for making estimates of other x and y values?

(x) Lot frontage (feet)	(y) Lot value ($)
70.8	10,100
60.0	9,000
85.0	12,700
75.2	11,120
69.5	11,000
84.0	12,500

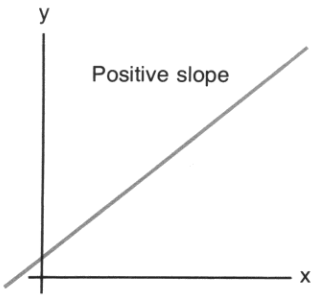
Accumulate the data using [Σ+]. Remember that when you enter two values, x and y, you must enter the y-value first.

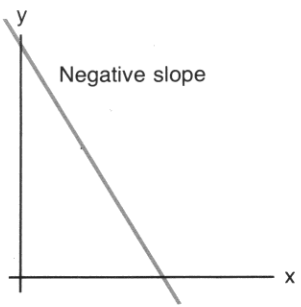
Keystrokes	Display	
[f] CLEAR [Σ]		
10100 [ENTER↑]	*10,100.00*	First entry.
70.8 [f] [Σ+]	*1.00*	
9000 [ENTER↑]	*9,000.00*	Second entry.
60 [f] [Σ+]	*2.00*	
12700 [ENTER↑]	*12,700.00*	Third entry.
85 [f] [Σ+]	*3.00*	
11120 [ENTER↑]	*11,120.00*	Fourth entry.
75.2 [f] [Σ+]	*4.00*	
11000 [ENTER↑]	*11,000.00*	Fifth entry.
69.5 [f] [Σ+]	*5.00*	
12500 [ENTER↑]	*12,500.00*	Sixth entry.
84 [f] [Σ+]	*6.00*	

Now that the data is stored, you can find the correlation coefficient by pressing [$\hat{x}$,r] or [$\hat{y}$,r], and then [x≷y].

Keystrokes	Display	
[g] [$\hat{y}$,r] [x≷y]	*0.97*	
or		
[g] [$\hat{x}$,r] [x≷y]	*0.97*	Correlation coefficient r.

The correlation coefficient, r, is always a value between −1 and +1. If r = +1, then the line has a positive (upward) slope and the data fits perfectly. If r = −1, the data still is a perfect fit but the line has a negative (downward) slope.





An example of a negative trend is declining property values or declining sales. If r = 0, the data values are spread out and do not come close to a straight line. It would be useless to find linear estimates from unrelated data.

In the example of lot frontage related to value, the correlation coefficient is close to 1, so we can feel comfortable using linear regression.

Suppose, though, that the correlation coefficient was not close to 1 but instead was 0.5 or 0.6. This would indicate that a straight line is not a very good fit to the data. Then you might try to fit a curve to the data.

Refer to the applications books for a description of three other types of curves: exponential, logarithmic, and power. A correlation coefficient can be calculated for each of these curves and should be interpreted similarly: if r is close to ±1, the curve is a reasonable approximation of the data. If not, try a different curve.

Linear Estimates $\hat{x}$, $\hat{y}$

Now that we know our data fits a line closely, we can trust results of our linear estimates. With the data totaled in registers R_1 through R_6, a predicted y (designated $\hat{y}$) can be calculated by keying in an x-value and pressing [ŷ,r]. And a predicted x (designated $\hat{x}$) can be calculated by keying in a y-value and pressing [x̂,r].

Example 1: For the previous example, find projected values for 80-, 95-, and 100-foot frontages.

Keystrokes	Display	
80 [g] [ŷ,r]	*11,922.65*	80-foot frontage projected value.
95 [g] [ŷ,r]	*14,084.29*	95-foot frontage projected value.
100 [g] [ŷ,r]	*14,804.83*	100-foot frontage projected value.

Given the value of a lot you could also estimate the lot frontage. What is the frontage of a lot valued at $14,500? At $12,000?

Keystrokes	Display	
14500 [9] [x̂,r]	*97.88*	Lot frontage (feet).
12000 [9] [x̂,r]	*80.54*	Lot frontage (feet).

If you wish to graph the regression line, you can calculate the coefficients of the linear equation $y = A + Bx$ as follows:

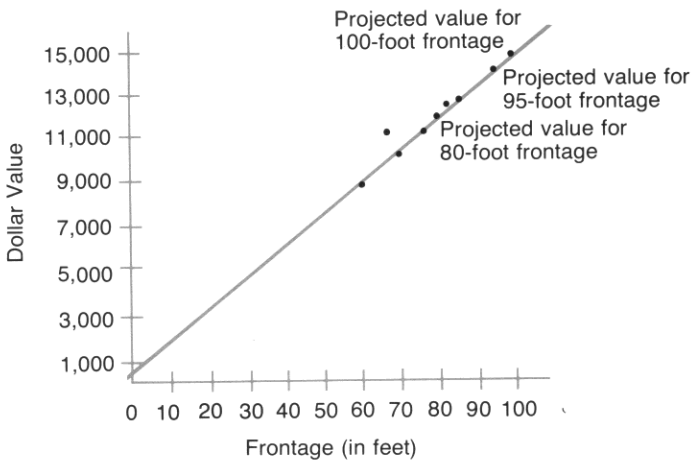
Keystrokes	Display	
0 [9] [ŷ,r]	*393.90*	y-intercept (A); projected value for x = 0.
[STO] 0	*393.90*	Store this value for calculating the slope (B).
0 [9] [x̂,r]	*-2.73*	x-intercept; when
[CHS]	*2.73*	y = 0 then x = −A/B.
[RCL] 0 [x≷y] [÷]	*144.11*	Slope of the line (B); indicates the change in the projected value caused by an incremental change in the x value.

Thus the equation that describes the regression line is:

$$y = 393.90 + 144.11\ x$$

You'll learn in the next pages how a key sequence like this can be simply put into a program so that pressing one key will compute both coefficients.

Plotting this example, you see that a 1-foot increase in the frontage results in a projected increase of $144.11 in value.



Example 2: You bought a house 3 years ago for $47,500. The first year it appreciated $5,000. The second year its value rose to $60,000. Today you figure the market price to be $64,000 if you were to sell. What will your house be worth next year?

Keystrokes	Display
f CLEAR Σ	
47500 ENTER↑	*47,500.00*
1 f Σ+	*1.00*
52500 ENTER↑	*52,500.00*
2 f Σ+	*2.00*
60000 ENTER↑	*60,000.00*
3 f Σ+	*3.00*
64000 ENTER↑	*64,000.00*
4 f Σ+	*4.00*

To make a projection for next year (year 5), simply solve for $\hat{y}$:

Keystrokes	Display
5 [9] [ŷ,r]	*70,250.00*

PART II
Programming

Section 5

What Is a Program?

A program is nothing more than a sequence of manual keystrokes that is remembered by the calculator. You can then execute the program as often as you like—typically with just one keystroke. The answer displayed at the end of execution is the same one you would have obtained by pressing the keys one at a time manually. No prior programming experience is necessary for HP-38E calculator programming.

As we briefly explained in the introduction, calculator programming is as simple as pressing the keys you would manually press to solve your problem. But even though HP-38E calculator programming is simple to understand and use, it is very powerful, featuring:

- An obvious programming language.
- Up to 99 lines of program memory.
- The ability to combine several keystrokes into each line.
- Decision-making capability for sophisticated routines.
- Several editing operations to facilitate corrections.

Together these features provide you with the tools necessary to tackle complex problems with unabashed confidence.

Why Write Programs?

Programs are written to save you time on repetitive calculations. Once you have written the keystroke procedure for solving a particular problem and recorded it in the calculator, you need no longer devote attention to the individual keystrokes that make up the procedure. You can let the calculator solve each problem for you. And because you can easily check the procedure in your program, you have more confidence in your final answer since you don't have to worry each time about whether or not you have pressed an incorrect key. The calculator performs the drudgery, leaving your mind free for more creative work.

Three Modes of Operation

There are three ways to use your HP-38E calculator:

1. Manual run mode.
2. Program mode.
3. Automatic run mode.

Manual Run Mode. The functions and operations you have learned about in the first part of this handbook are performed manually one at a time. These functions combined with the automatic memory stack enable you to calculate any problem with ease.

Program Mode. In *program* mode the functions and operations you have learned about are not executed, but instead are recorded in a part of the calculator called *program memory* for later execution. To get into *program* mode, simply press [9] [P/R] *(program/run)*. All operations on the keyboard except the following six* can be recorded for later execution when the calculator is in *program* mode. These six operations cannot be recorded:

[P/R]	[SST]
[MEM]	[BST]
[CL P]	[GTO] [•]

These six operations work in *program* mode to help you write and record your programs.

Automatic Run Mode. The HP-38E can also be used to automatically execute a list of operations when the calculator is in *run* mode if they have previously been recorded in program memory. Instead of pressing each key manually, the recorded operations are executed sequentially in automatic *run* mode when you press [R/S] *(run/stop)*. You press only one key and the entire list of recorded operations is executed much more quickly than you could have executed them yourself.

---

*The following operations are also not recordable but they do not assist you in programming: [f] CLEAR [ALL], [f] CLEAR [PREFIX], [STO] [ENTER↑]. The first two operations do nothing in *program* mode. [STO] [ENTER↑] will perform the self-check, which clears the automatic memory stack in The HP-38C and *all* memory in the HP-38E.

Introductory Program

The discount price plus sales tax program you wrote, recorded, and executed in the introduction showed you that the sequence of keystrokes used to solve a problem manually is the same sequence used in a program. Now let's return our attention to that program to explain the information displayed in *program* mode.

First put the calculator in *program* mode by pressing [g] [P/R] so that the next sequence of keystrokes will be recorded for later execution. Second, press [g] [CL P] to clear the calculator of previous programs. The display will show:

00-

This tells you that you are at the beginning of program memory. Line 00 contains an automatic stop instruction and cannot be used to record your program keystrokes. Program keystrokes are recorded in lines 01 through 99.

With *00-* displayed in program mode, you are ready to key in your program. The short list of keys for calculating 25% discount on cost plus 6% sales tax is:

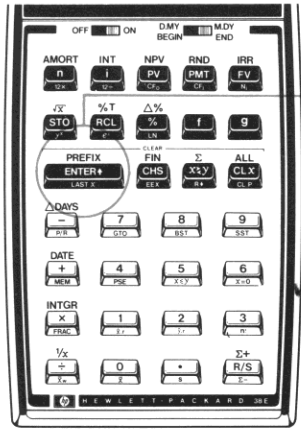
Keys	Comments
[ENTER↑]	This key separates the cost from the discount rate.
25 [%] [−]	These keys calculate the amount of discount and subtract this amount from the original marked price.
6 [%] [+]	These keys calculate the sales tax and add it to the discounted cost.

Keycodes

Press the first key of the program and the display will change to:

01- 31

The two digits displayed on the left designate the line number of program memory, while the two digits displayed on the right designate the key stored in that line. Each key on the keyboard has a two-digit keycode, except for the digit keys. For convenience, the digit keys, and their respective alternate functions, are coded 0 through 9. All other keys are coded by their position on the keyboard. The first digit denotes the row of the key and the second digit the number of the key in that row. So *31* tells you that the key is in the third row on the calculator and that it is the first key in that row, the [ENTER↑] key.



Third row
First key

This handy matrix system allows you to easily determine the code for each instruction without using a reference table.

The keys for finding a 25% discount on cost plus 6% sales tax and their corresponding displays are shown below. Press each key in turn and verify the keycode shown in the display.

Key	Display	
[ENTER↑]	*01-*	*31*
2	*02-*	*2*
5	*03-*	*5*
[%]	*04-*	*23*
[−]	*05-*	*41*
6	*06-*	*6*
[%]	*07-*	*23*
[+]	*08-*	*51*

Running a Program

Programs are executed in automatic *run* mode. To return to *run* mode from *program* mode, press [g] [P/R] *(program/run)*. Returning to *run* mode automatically resets the calculator so that program execution will begin from line 00. (Pressing [g] [CL P] in *run* mode accomplishes the same thing.) Then, key in a value for the cost of an item and press [R/S] *(run/stop)* in *run* mode to run your program. The operations stored in program memory are executed sequentially downward from line 00. First line 01 is executed, then line 02, then line 03, and so on up through line 08 which contains the last step in program memory. After executing the last step in program memory (or a [GTO] 00 program instruction) the calculator returns to line 00 and stops.

[GTO] 00

The calculator automatically maintains a [GTO] 00 instruction in every unused line of program memory. If you clear program memory by pressing [g] [CLP] in program mode,* memory is filled with [g] [GTO] 00. The eight-line program you keyed in replaced eight of these instructions.

Notice that all four keys composing the [g] [GTO] 00 instruction are merged into one line of the program to conserve program memory. (You will see more examples of merged keycodes later.)

When you keyed in your program, the program memory changed...

...from this...	...to this.
00–	00–
01–25 7 00	01– 31
02–25 7 00	02– 2
03–25 7 00	03– 5
04–25 7 00	04– 23
05–25 7 00	05– 41
06–25 7 00	06– 6
07–25 7 00	07– 23
08–25 7 00	08– 51
09–25 7 00	09–25 7 00

* Or, in the HP-38*E* only, when program memory is cleared by turning the calculator off, then on.

The illustration on the left shows the first 8 lines of program memory immediately after pressing [g] [CLP] in program mode. The illustration on the right shows program memory after recording the eight-line example program.

A [GTO] 00 instruction in the program tells the calculator to go to line 00 and execute the automatic stop instruction there next. If [R/S] is pressed again in automatic run mode, the calculator will begin executing instructions from line 00 as it did the first time. Each time the calculator executes this program, it ends execution at line 00, ready to begin again.

If you had recorded a 99-line program, after executing line 99 the calculator would go to line 00, the automatic stop instruction, and stop. Then you would have to press [R/S] to execute the program again.

Now try an example.

Example: Calculate the discounted price plus sales tax of a lamp marked $75 and an ottoman marked $115. (Remember, we are still assuming a 25% discount and 6% sales tax.)

Keystrokes	Display	
75 [R/S]	*59.63*	Discounted price of lamp plus sales tax.
115 [R/S]	*91.43*	Discounted price of ottoman plus sales tax.

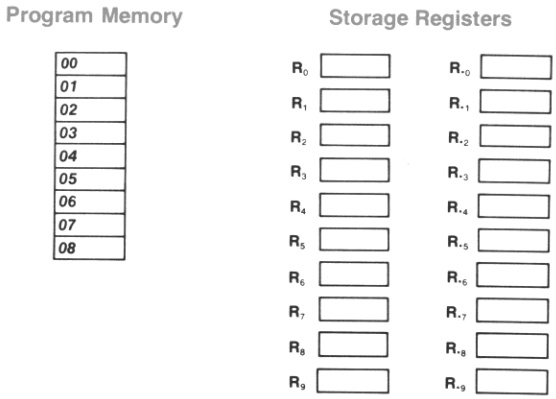
When you hold down the [R/S] key, the display will show the line where program execution begins.

Each time you press [R/S] the calculator executes the sequence of keystrokes you have recorded. You calculate the same answers you would obtain if you did each problem manually, but without the time or the tedium.

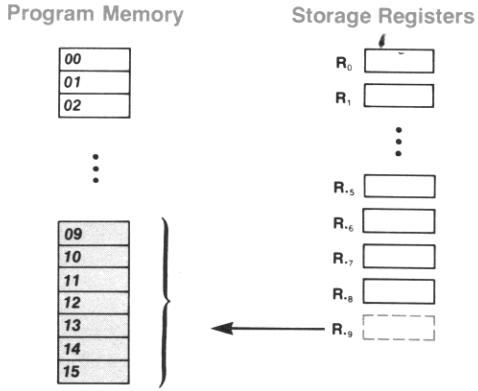
Automatic Memory Allocation

When you clear the calculator of previous programs, you have eight lines of program memory available and 20 storage registers. The memory is

allocated like this:



As you key in the ninth line of programming, storage register R.9 is converted to seven lines of program memory so that now the memory allocation looks like this:



Each time you use up seven more lines of programming, the calculator converts the bottom storage register into program memory.

There is a function, [MEM] *(memory map)*, on your calculator which describes the current memory allocation at any time, in or out of program mode. When you press [g] [MEM] after the program memory has been cleared, the display will show:

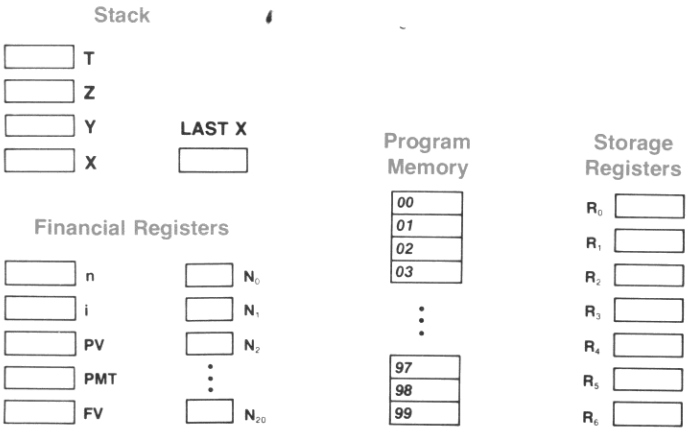
P-08 r-20

Available program lines **Available storage registers**

As long as you are pressing [MEM] the memory allocation will be displayed. When you let go of the [MEM] key, the calculator returns to the original display. So at any time, you can find out the number of lines available for programming and the number of registers available for storing data.

You don't need to worry about the memory allocation if you're using only the first seven storage registers—the calculator converts storage registers to program memory automatically but never converts registers R_0 through R_6. The [MEM] function is extremely useful when you have recorded a lengthy program and you want to store data in more than seven registers. You know the exact amount of memory allotted for each.

When you record a full 99 lines of program memory, a map of the calculator's memory registers looks like this:



As you can see, program memory is separate from the four stack registers, the LAST X register, and the financial registers. Notice that instead of the original 20 storage registers (R_0 through R_9, $R._0$ through $R._9$) we now have seven storage registers, all of which will allow you to perform storage register arithmetic. R_1 through R_6 also accumulate statistical data.

What happened to storage registers R_7 through R_9 and $R._0$ through $R._9$? They have been converted to program memory. Each storage register is worth seven lines of programming. Rather than give you all 99 lines of programming memory at once, thus depriving you of 13 storage registers regardless of program length, the calculator converts storage registers into program memory one at a time, as you need them.

Automatic Memory Allocation

After the initial eight lines of programming, the calculator converts storage registers to program memory, seven lines at a time.

When you have P-lines of program memory, you have r-storage registers left.

P-08	*r-20*
P-15	*r-19*
P-22	*r-18*
P-29	*r-17*
P-36	*r-16*
P-43	*r-15*
P-50	*r-14*
P-57	*r-13*
P-64	*r-12*
P-71	*r-11*
P-78	*r-10*
P-85	*r-09*
P-92	*r-08*
P-99	*r-07*

Section 6:

Writing a Second Program

Now let's write a second program and use it to further explore the programming capability of your calculator. Suppose you are a jewelry distributor and you want to write a program that will help you complete your billing invoices.

You've received the following order, and need to complete the last three columns of the invoice.

DIRECT FORM PURCHASE REQUISITION ORDER

P.O. No. 25-

RASTON, UNGER, BENTZ & YATES
JEWELERS
2561 N.W. Morrison Ave.
New York, New York 14203
Telephone (716) 731-8240

ORDER DATE	CONFIRMING	SHIP VIA SURFACE ☐ AIR ☐ UPS ☐ VENDOR ☐ WILL CALL ☐ OTHER

ITEM	QTY.	DESCRIPTION	UNIT PRICE	TOTAL	TAX 6.75%	FINAL
1	13	SS4 Star Sapphire	$ 68.50	?	?	?
2	18	RG13 Ruby Ring	72.90	?	?	?
3	24	GB87 Gold Band	85.00	?	?	?
4	5	DG163 Diamond	345.00	?	?	?
5	9	DF8 Fire Opal	155.25	?	?	?
6						

For each line of the invoice, you need to calculate the total (quantity × unit price), 6.75% tax on the total, and final cost (total + tax).

Before you begin, store the tax rate in storage register R_0. This will decrease the number of keystrokes used to solve the problem.

Keystrokes	Display	
6.75 [STO] 0	*6.75*	Store tax rate.

If you enter the quantity and key in the unit price for each line, the problem could be solved manually by pressing the keys shown in the left-hand column that follows. Let's complete the first line of the invoice, manually:

Keystrokes	Display	
13 [ENTER↑] 68.5	*68.5*	Enter quantity and key in unit price.
[×]	*890.50*	TOTAL.
[RCL] 0	*6.75*	Recall tax rate.
[%]	*60.11*	TAX (6.75% of $890.50).
[+]	*950.61*	FINAL ($890.50 plus $60.11).

The program keystrokes for this problem are the same, only we begin with the [×] key so that we can enter different quantities and unit prices. Simply toggle into *program* mode by pressing [g] [P/R] and press [g] [CL P] to clear program memory and display line 00. Then key in the list of keystrokes above, omitting 13 [ENTER↑] 68.5. Verify that each keycode is correct as you key in each instruction by checking the displays shown.

Keystrokes	Display	
[×]	*01- 61*	Quantity times price.
[RCL] 0	*02- 22 0*	Recall tax rate.
[%]	*03- 23*	% tax of total quantity.
[+]	*04- 51*	Add for net cost.

The keys are not executed, but are recorded in program memory lines 01 through 04.

Notice that you recorded two keystrokes, [RCL] 0, into one line (02) of the program. Earlier, we saw that the four keys in [g] [GTO] 00 were also recorded in one line of programming. [RCL] 0, like [g] [GTO] 00, has a merged keycode to conserve program memory. In every case where several keys are required to perform a single operation (e.g. [f] [%T], [STO] [+] 1, [g] [CF_0]), the keycodes are merged into one line of program memory.

> **Note:** Each operation, prefixed or not, requires only one line of program memory.

For each keycode, the two digits on the left of the display designate the line number of program memory that is being displayed. The rest of the numbers in the display indicate the keys used to perform one operation. For instance, 02 above indicates that [RCL] 0 has been recorded in line 02 of program memory. The digits 22 denote the [RCL] key and the digit 0 denotes the 0 key. The operation stored then, is [RCL] 0 which recalls the number in storage register R_0 into the display or X-register.

Now, to run the program we have just recorded, toggle back into *run* mode and enter the quantity and unit price of the first item on the invoice again to compare with the previous results. Then press [R/S] to execute the program.

Keystrokes	**Display**	
[g] [P/R]		Toggle back into *run* mode.
13 [ENTER↑] 68.5	*68.5*	Enter quantity and key in price.
[R/S]	*950.61*	FINAL (net amount). What happened?

The program didn't stop to show the intermediate results for the total and tax columns. The calculator executed each program line sequentially until it reached the end of the program. Thus, only the final result is displayed. But that's easy to change. First, we'll discuss some keys that will aid you in reviewing your programs.

Displaying Each Line

In order to look at this program, you need to be able to display each step. Two operations allow you to do this: [SST] *(single step)* and [BST] *(back step)*.

With the invoice program still recorded in the calculator, toggle into *program* mode by pressing [g] [P/R] and press [g] [SST] once. The display will change to:

01- 61

Press [g] [SST] again and the display will change to:

02- 22 0

Now press [g] [BST]. You can see what has happened. You are back at program memory line 01. Press [g] [BST] again and line 00 is displayed.

So [SST] single steps to display the contents of the next line of program memory.

And [BST] backsteps to display the contents of the previous line of program memory.

Of course, because these two keys work in *program* mode, neither can be stored in program memory.

Displaying a Particular Line

If you want to see one of the later lines in a lengthy program, [SST] is inconvenient, just as backstepping through eight or 80 lines of programming would be annoying. To display a particular line of program memory use the [GTO] instruction *followed by a decimal point* and then key in the desired two-digit line number.

> **Remember:** To display a line in program mode, press the decimal point before you designate the line number of the [GTO] instruction. If you do not use the decimal point, your [GTO] instruction will be recorded for program execution.

When you are in *run* mode, you can specify the line number in a [GTO] instruction either way; with or without a decimal point. The display remains unchanged, but program memory will be set to the line number specified (as long as that line number is currently allocated to program memory).

For example, to see line 04 in the previous program, press [g] [GTO] .04 so that the display shows:

04- 51

When using the [GTO] key in this way, always use two digits for designating line numbers (01 through 99). If you attempt to go to a line number that is not within memory as currently described by [g] [MEM] the calculator will display *Error 4*.

Try to go to line 56 of program memory:

Keystrokes	Display
[g] [GTO] .56	*Error 4*

This means that you have attempted to go outside the current allocation for program memory. You saw earlier that the calculator gives you eight program lines to start writing your program. Don't worry—if you have a long program and really do need to see line 56, the calculator will let you do it.

Now, pressing any key will clear the error message from the calculator and return it to the original display.

Interrupting Program Execution

From time to time you will want a program to stop execution by itself so that you can enter new data or view an intermediate result. (In our invoice example, we would like to see the intermediate results for TOTAL and TAX so that they may be recorded.) There are two operations on your HP-38E calculator that will automatically interrupt program execution when they are encountered as program instructions: [R/S] *(run/stop)* and [PSE] *(pause)*.

Stopping Program Execution

[R/S] works differently as an executed instruction in a program than it does when pressed from the keyboard in *run* mode. As an executed instruction, [R/S] stops program execution, allowing you to key in new data or to write down an intermediate result.

When [R/S] is then pressed from the keyboard in *run* mode, the calculator resumes program execution sequentially downward.

Let's modify our invoice program to see how [R/S] can be used (and to complete our invoice!). We want to record [R/S] in two places.

To record the program, make sure that you are in program mode (a line number will be in the display on the left) and press [g] [CL P] to erase our previous program and display line 00. Then key in the following list of keys.

Keystrokes	Display		
[×]	01–	61	Quantity times unit price.
[R/S]	02–	74	Stop to record the TOTAL.
[RCL] 0	03–	22 0	Recall tax rate from R_0.
[%]	04–	23	Calculate tax on TOTAL.
[R/S]	05–	74	Stop to record the TAX.
[+]	06–	51	FINAL (net amount TAX + TOTAL).

In order to run this program, toggle back into *run* mode by pressing [g] [P/R] Now use the program to complete the invoice below. (You may want to check storage register R_0 to see if the 6.75% tax rate is still there. Press [RCL] 0. If 6.75 appears on the display, you haven't previously erased it. If not, key in 6.75 and press [STO] 0.)

DIRECT FORM PURCHASE REQUISITION ORDER

P.O. No. 25-

RASTON, UNGER, BENTZ & YATES
JEWELERS
2561 N.W. Morrison Ave.
New York, New York 14203
Telephone (716) 731-8240

ORDER DATE	CONFIRMING	SHIP VIA SURFACE □ AIR □ UPS □ VENDOR □ WILL CALL □ OTHER

ITEM	QTY.	DESCRIPTION	UNIT PRICE	TOTAL	TAX 6.75%	FINAL
1	13	SS4 Star Sapphire	$ 68.50	$890.50	$60.11	$950.61
2	18	RG13 Ruby Ring	72.90	?	?	?
3	24	GB87 Gold Band	85.00	?	?	?
4	5	DG163 Diamond	345.00	?	?	?
5	9	DF8 Fire Opal	155.25	?	?	?

Let's start with the first item on the invoice again, so that you can verify the answers obtained manually.

Keystrokes	Display	
13 [ENTER↑] 68.5	*68.5*	Enter quantity and key in unit price.
[R/S]	*890.50*	Program stops to display TOTAL.
[R/S]	*60.11*	Program stops to display TAX.
[R/S]	*950.61*	FINAL net amount of first item is calculated.
18 [ENTER↑] 72.9	*72.9*	Enter quantity and price of second item.
[R/S]	*1,312.20*	TOTAL.
[R/S]	*88.57*	TAX.
[R/S]	*1,400.77*	FINAL.
24 [ENTER↑] 85	*85.*	Third item.
[R/S]	*2,040.00*	TOTAL.
[R/S]	*137.70*	TAX.
[R/S]	*2,177.70*	FINAL.
5 [ENTER↑] 345	*345.*	Fourth item.
[R/S]	*1,725.00*	TOTAL.
[R/S]	*116.44*	TAX.
[R/S]	*1,841.44*	FINAL.
9 [ENTER↑] 155.25	*155.25*	Fifth item.
[R/S]	*1,397.25*	TOTAL.
[R/S]	*94.31*	TAX.
[R/S]	*1,491.56*	FINAL.

With the quantity in the Y-register and the unit price in the X-register, pressing [R/S] in *run* mode calculates the total (quantity × price) and stops at the first [R/S] instruction encountered. Pressing [R/S] again calculates the amount of tax on the total quantity. Pressing [R/S] a third time calculates the final net amount for the item and program execution stops at line 00, ready to run again.

In general, [R/S] is recorded into a program when you need to display more than one answer. As we have seen, [R/S] is not necessary when

you want to display *only one* answer or the *final* answer of a series since the calculator automatically ends execution at line 00, ready to begin again.

Pausing During Program Execution

A [PSE] *(pause)* instruction executed in a program momentarily interrupts program execution to display intermediate results that do not have to be written down. The length of the pause is about 1 second, although more than one [PSE] instruction can be used to lengthen the time if desired.

To see how [PSE] can be used in a program, we'll modify the invoice program again. In the new program, the TOTAL and TAX amounts will be briefly displayed before the FINAL answer is calculated.

At the same time, it would be convenient to sum all three of the last columns on the invoice, so that we don't have to compute their sums manually. We will incorporate storage register arithmetic (refer to *Your HP Financial Calculator,* The Display and Memory) into the program so that the answers will be at your fingertips when we're done.

Before you key in the program, press [f] CLEAR [Σ], to clear the statistical registers R_1 through R_6. (We will use three of these registers for summing.) The 6.75 tax rate should still be in storage register R_0.

Now press [g] [P/R] to toggle into *program* mode. Then press [g] [CL P] to clear program memory and display line 00. Key in the following list of keys.

Keystrokes	Display	
[×]	*01- 61*	Multiplies quantity times price.
[g] [PSE]	*02- 25 4*	Pauses to display TOTAL.
[STO] [+] 1	*03-21 51 1*	Adds TOTAL to contents of R_1.
[RCL] 0	*04- 22 0*	Recalls tax rate.
[%]	*05- 23*	Finds % tax of TOTAL.
[g] [PSE]	*06- 25 4*	Pauses to display TAX.
[STO] [+] 2	*07-21 51 2*	Adds TAX to contents of R_2.
[+]	*08- 51*	Adds TOTAL to TAX.
[STO] [+] 3	*09-21 51 3*	Adds FINAL to contents of R_3.

Now toggle back into *run* mode by pressing [9] [P/R]. This program also assumes that you enter the quantity into the Y-register and the price into the X-register.

DIRECT FORM PURCHASE REQUISITION ORDER

P.O. No. 25-

RASTON, UNGER, BENTZ & YATES
JEWELERS
2561 N.W. Morrison Ave.
New York, New York 14203
Telephone (716) 731-8240

ORDER DATE	CONFIRMING	SHIP VIA SURFACE ☐ AIR ☐ UPS ☐ VENDOR ☐ WILL CALL ☐ OTHER			

ITEM	QTY.	DESCRIPTION	UNIT PRICE	TOTAL	TAX 6.75%	FINAL
1	13	SS4 Star Sapphire	$ 68.50	$890.50	$60.11	$950.61
2	18	RG13 Ruby Ring	72.90	1,312.20	88.57	1,400.77
3	24	GB87 Gold Band	85.00	2,040.00	137.70	2,177.70
4	5	DG163 Diamond	345.00	1,725.00	116.44	1,841.44
5	9	DF8 Fire Opal	155.25	1,397.25	94.31	1,491.56
6				?	?	?

Keystrokes	Display	
13 [ENTER↑] 68.5		Item # 1.
[R/S]	*890.50*	TOTAL.
	60.11	TAX.
	950.61	FINAL.
18 [ENTER↑] 72.9		Item # 2.
[R/S]	*1,312.20*	TOTAL.
	88.57	TAX.
	1,400.77	FINAL.

Keystrokes	Display	
24 [ENTER↑] 85		Item #3.
[R/S]	*2,040.00*	TOTAL.
	137.70	TAX.
	2,177.70	FINAL.
5 [ENTER↑] 345		Item #4.
[R/S]	*1,725.00*	TOTAL.
	116.44	TAX.
	1,841.44	FINAL.
9 [ENTER↑] 155.25		Item #5.
[R/S]	*1,397.25*	TOTAL.
	94.31	TAX.
	1,491.56	FINAL.
[RCL] 1	*7,364.95*	Sum of TOTAL amounts.
[RCL] 2	*497.13*	Sum of TAX amounts.
[RCL] 3	*7,862.08*	FINAL sum.

Program Stops

At times a mistake of some kind in your program will stop program execution. To help you identify why the calculator stopped in the middle of your program, possible reasons are listed below.

Executing a R/S. The execution of a [R/S] instruction in a program halts program execution at the line following the [R/S].

Executing Line 00. Whenever line 00 is executed in a program, program execution stops at line 00.

Pressing Any Key. Pressing any key halts program execution. You can press [R/S] again to resume program execution from where it stopped. If you toggle into *program* mode to see where the program stopped, be sure to note the line number. To resume program execution from that line, toggle back into *run* mode and use a [GTO] instruction to go to that line number before pressing [R/S].

Overflow Calculations. Your calculator has been designed so that you can always tell why the calculator stops by looking at the display. If program execution stops because the result of a calculation in the X-register is a number with a magnitude greater than $9.999999999 \times 10^{99}$, all 9's are displayed with appropriate sign. It is then easy to determine the operation that caused the overflow by toggling to *program* mode and identifying the keycode in the display.

If the overflow occurs in one of the storage registers as the result of storage register arithmetic, the calculator will display *Error 1* to inform you of the overflow.* Check the storage registers to see in which register the overflow has occurred.

Error Stops. If the calculator attempts to execute any error-causing operation during a running program, execution immediately halts and the calculator displays *Error* and a number from 1 to 8. Press any key to clear the error message (the function is not executed) and toggle into *program* mode to see the line and keycode of the error-causing instruction. Don't forget, when you toggle back into *run* mode, you'll be back at line 00.

* In the HP-38*E* only, ***Error 1*** will also result from a storage register overflow caused by summation with [Σ+].

Section 7:

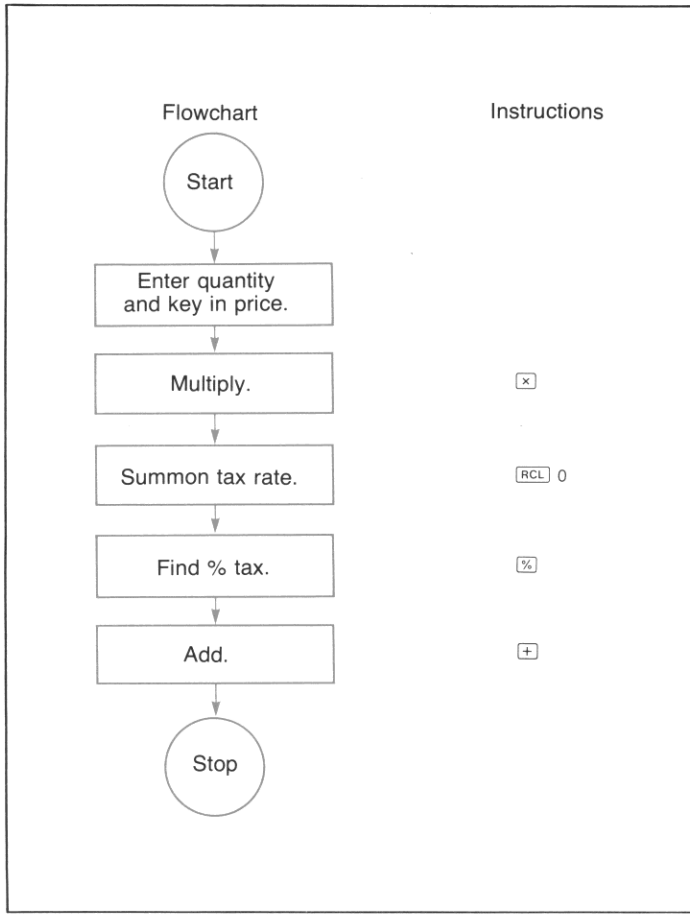
Decisions

Flowcharts

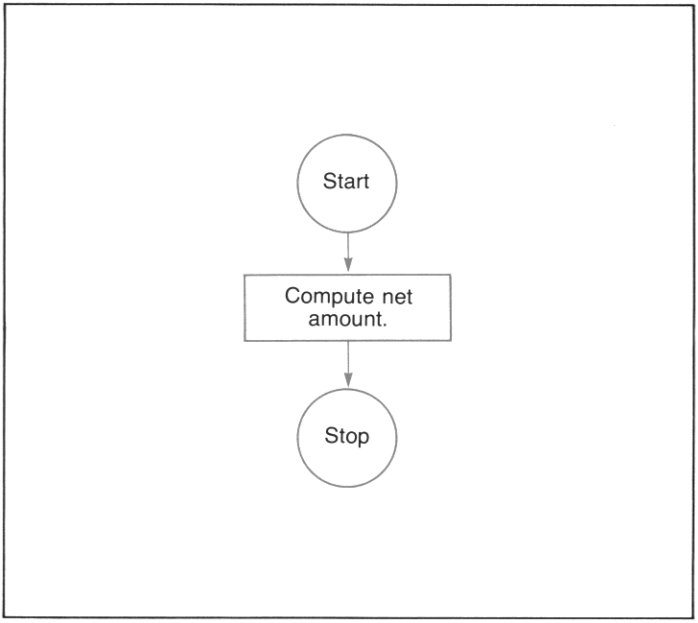
At this point, we digress for a moment from our discussion of the calculator itself to familiarize ourselves with a fundamental and extremely useful tool in programming—the flowchart.

A flowchart is an *outline* of the way a program solves a problem. With 99 possible instructions, it is quite easy to get "lost" while creating a long program, especially if you try to simply load the complete program from beginning to end with no breaks. A flowchart is a shorthand that can help you design your program by breaking it down into smaller groups of instructions. It is also very useful as documentation—a road map that summarizes the operation of a program.

A flowchart can be as simple or as detailed as you like. Here is a flowchart that shows the operations you executed to find the total, tax, and final amounts on an invoice. Compare the flowchart to the actual instructions for the program.



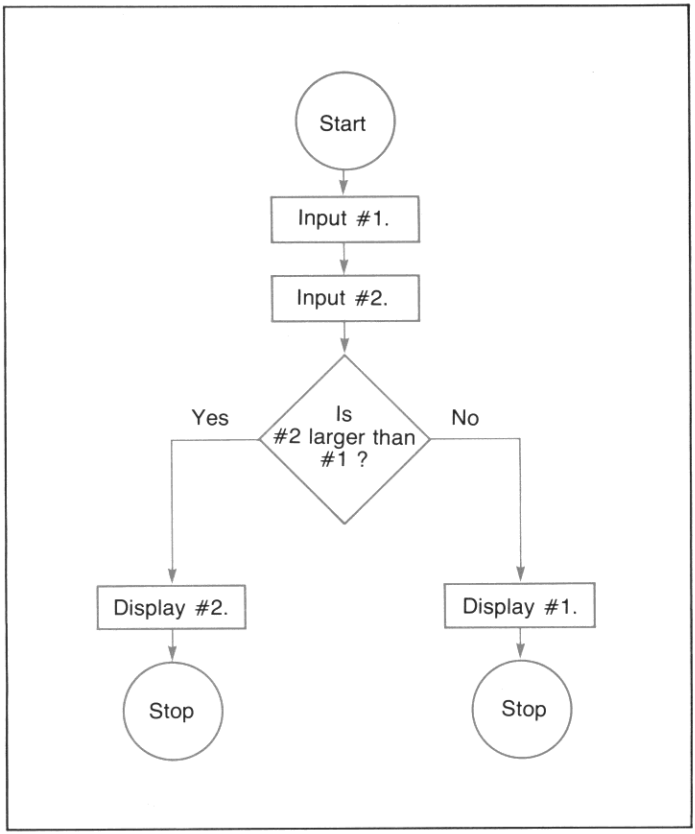
You can see the similarities. At times, a flowchart may duplicate the set of instructions exactly, as shown above. At other times, it may be more useful to have an entire group of instructions represented by a single block in the flowchart for the program to calculate the net amount of an item or quantity.



Here an entire group of instructions was replaced by one block in the flowchart. This is a common practice, and one which makes a flowchart extremely useful in visualizing a complete program.

You can see how a flowchart is drawn linearly, from the top of the page to the bottom. This represents the general flow of the program, from beginning to end. Although flowcharting symbols sometimes vary, throughout this handbook we have held to the convention of circles for the beginning and end of a program or routine, and rectangles to represent groups of functions that take an input, process it, and yield a single output. We have used a diamond to represent a *decision,* where a single input can yield either of two outputs.

For example, if you had two numbers and wished to write a program that would display only the larger, you might design your program by first drawing a flowchart that looks like the one shown on the following page:



After drawing the flowchart, you would go back and substitute groups of instructions for each element of the flowchart. When the program was loaded into the calculator and run, if # 2 was larger than # 1, the answer to the question "Is # 2 larger than # 1?" would be YES, and the program would take the left-hand path, display #2, and stop. If the answer to the question was NO, the program would execute the right-hand path, and # 1 would be displayed. (You will see later the decision-making instructions available on your calculator.)

As you work through this handbook, you will become more familiar with flowcharts. Use the flowcharts that illustrate the examples and problems to help you understand the many features of the calculator, and draw your own flowcharts to help you create, edit, eliminate errors in, and document your programs.

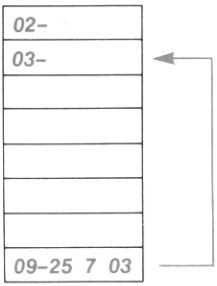
Branching

Although program execution is normally sequential, with one line executed after another, execution can be transferred or "branched" to any line in program memory. The "branch" can be made unconditionally or it can be made dependent on the outcome of a comparison of data values.

Unconditional Branching and Looping

You have seen how GTO . is used in program mode to help you *display* any line in program memory. You can also use the GTO instruction as part of a program, to *branch* program execution to the line number specified. It can tell the calculator to execute line 00 next, as we have already seen, or to execute any other line in program memory.

By using a GTO instruction in your program, you can transfer execution to any part of the program that you choose.



Execution branches to line 03.

(g GTO 03)

A [GTO] instruction used this way is known as an *unconditional branch*. It always *branches* execution from the [GTO] instruction to the specified line number.

When recording a "go to" instruction, always follow [GTO] with two digit keys to designate the line number. For instance, to branch to line 3, the program instruction must be [g] [GTO] 03.

A common use of a branch is to create a "loop" in a program. For example, the following program calculates the squares (a number times itself) of consecutive integers beginning with zero. The calculator continues to compute the square of the next consecutive integer until you press [R/S] to stop program execution. The simple formula used is: $x = n \times n$ where n is continually incremented by one.

To key in the program, toggle into *program* mode by pressing [g] [P/R]

Keystrokes	**Display**	
[g] [CL P]	*00-*	Clear program memory.
0	*01- 0*	
[STO] 1	*02- 21 1*	
[RCL] 1	*03- 22 1*	
[g] [PSE]	*04- 25 4*	Pause to display integer.
[ENTER↑]	*05- 31*	
[×]	*06- 61*	
[g] [PSE]	*07- 25 4*	Pause to display square.
1	*08- 1*	
[STO] [+] 1	*09-21 51 1*	
[g] [GTO] 03	*10-25 7 03*	

To run the program, press [g] [P/R] and then press [R/S]. The program will begin displaying integers and their squares and will continue to do so until you press [R/S] from the keyboard or until the calculator overflows.

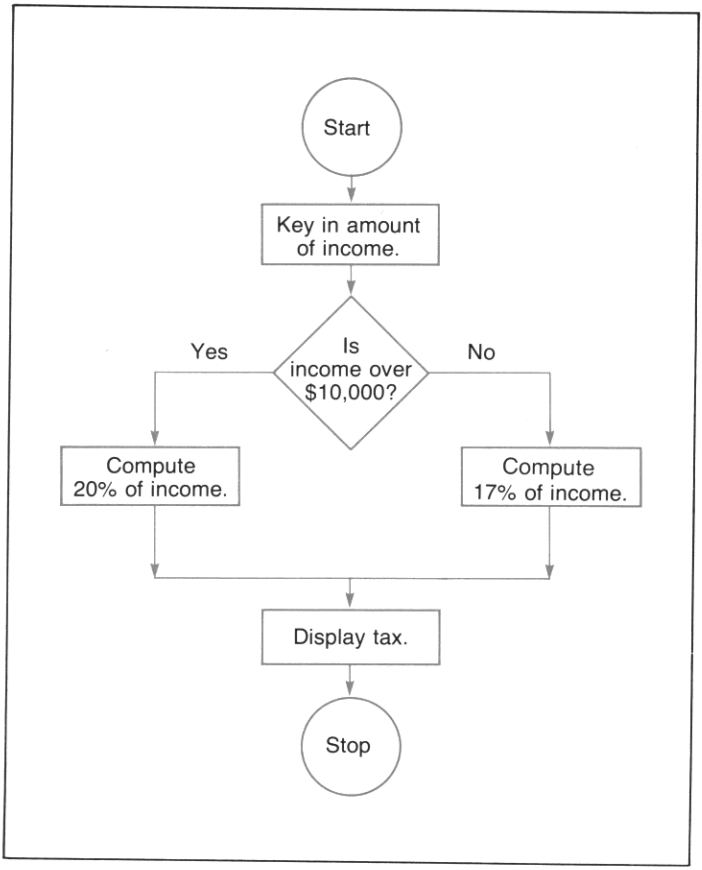
How it works: The GTO 03 instruction causes the calculator to break the sequential execution and resume from the new line.

Since execution is transferred to the instruction in line 03 each time the calculator encounters the GTO 03 instruction in line 10, the calculator will remain in this "loop," continually adding one to the number in storage register R_1 and displaying the new number and its square.

Looping techniques like the one illustrated here are common and extraordinarily useful in programming. By using loops, you take advantage of one of the most powerful features of your calculator—the ability to update data and perform calculations automatically, quickly, and if you so desire, endlessly.

Conditional Tests and Conditional Branching

Often there are times when you want a program to make a decision. For example, suppose an accountant wishes to write a program that will calculate the amount of tax to be paid by a number of persons. For those with incomes of $10,000 or less, the tax is 17%. For those with incomes of more than $10,000, the tax is 20%. A flowchart for the program might look like this:

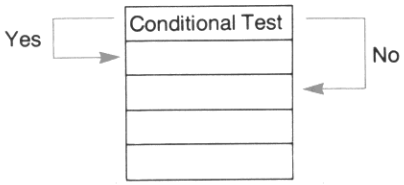


Two different program instructions enable your calculator to make decisions, like the one shown above, depending on the outcome of a comparison of data values.

[g] [x≤y] Tests to see if the value in the X-register is less than or equal to the value in the Y-register.

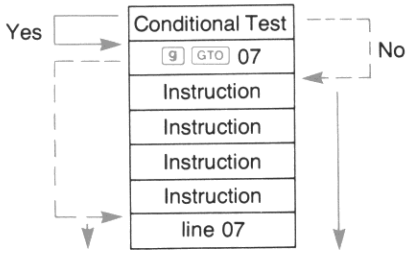
[g] [x=0] Tests to see if the value in the X-register is equal to zero.

Each conditional essentially asks a question when it is encountered as an instruction in a program. If the answer is YES, program execution continues sequentially downward with the next instruction in program memory. If the answer is NO, the calculator branches *around* the next instruction. For example:



You can see that after it has made the conditional test, the calculator will do the next instruction if the test is *true*. This is the "DO if TRUE" rule.

The line immediately following the conditional test can contain any instruction. The most commonly used instruction will be a [GTO] instruction. This will branch program execution to another section of program memory if the conditional test is true.



Now let's look at that accountant's problem again. For persons with incomes of more than $10,000, the tax is 20%. For persons with incomes of $10,000 or less, the tax is 17%. The following program will test the amount in the X-register and compute the correct amount of tax.

Keystrokes	Display	
10000 STO 0	*10,000.00*	Store test value in R_0 while still in *run* mode.
g P/R		Toggle into *program* mode.
g CL P	*00–*	Clear program memory.
RCL 0	*01– 22 0*	Recall contents of R_0...
x≷y	*02– 33*	and place in Y-register.
g x≤y	*03– 25 5*	If income less than or equal
g GTO 08	*04–25 7 08*	to $10,000, go to line 08; if
2	*05– 2*	not, skip to line 05.
0	*06– 0*	
g GTO 10	*07–25 7 10*	
1	*08– 1*	
7	*09– 7*	
%	*10– 23*	

Now let's run the program to compute taxes on incomes of $15,000 and $7,500:

Keystrokes	Display	
g P/R		Back to *run* mode.
15000 R/S	*3,000.00*	Tax.
7500 R/S	*1,275.00*	Tax.

The program discussed earlier in this section, calculating the squares of integers, contained an *infinite* loop. In other words, the calculator would continue to compute squares of integers until you stopped it. But you can use the decision-making power of your calculator to get out of a loop.

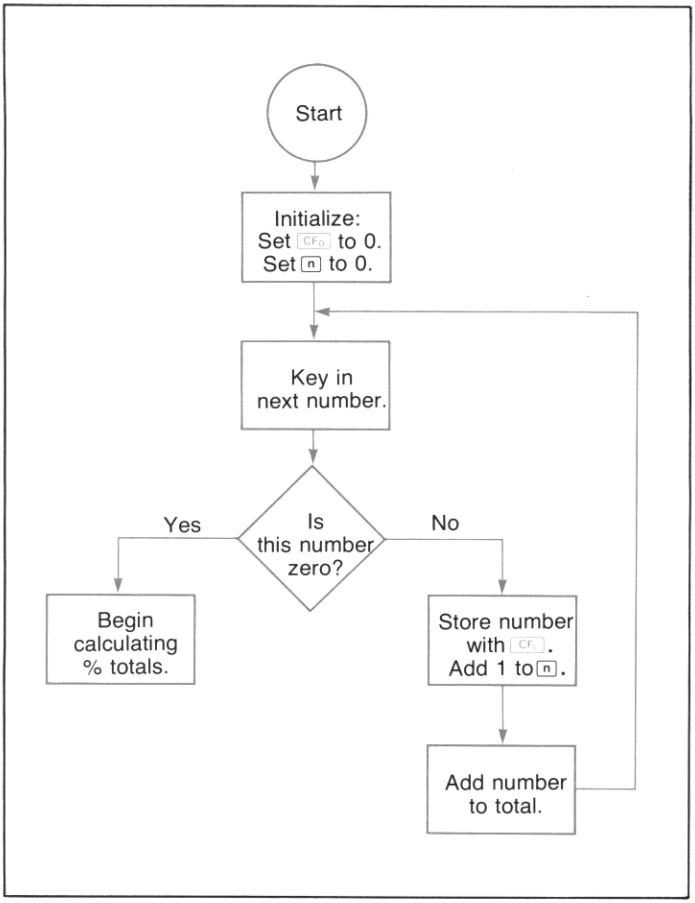
Let's write a program to calculate the percentage that each number in a series of numbers is of the total sum of the numbers. We will use the CF_0 and CF_j keys to input the numbers rather than STO because, as we have seen in the discounted cash flow analysis section, CF_0 not only stores a number in storage register R_0, but also sets the n register to zero. The CF_j key stores numbers in consecutive storage registers R_1, R_2, R_3, ... and also increments the contents of the n register by one, each time it is pressed.

This program will consist of two main parts:

1. Input the series of numbers and find their sum.
2. Calculate the %T for each number of that sum.

To input the numbers using a program, we will want to use a flag of some sort to tell the calculator that we have finished entering the series of numbers. Since x=0 is a conditional on the HP-38E, let's use zero as a flag. This means that when the calculator reads a zero entry, it will branch to the second part of the program.

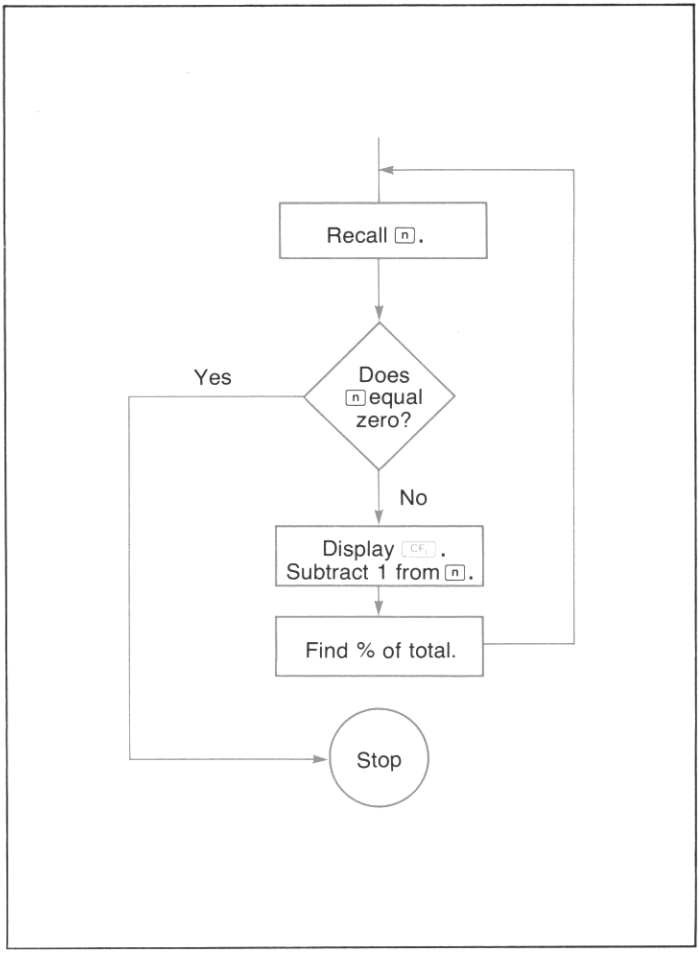
The flowchart for the first half of the program looks like this:



Now let's record the first part of the program:

Keystrokes	Display	
g P/R	*00–*	Toggle into *program* mode.
g CL P	*00–*	Clear program memory.
g CF_0	*01– 25 13*	Store displayed number in R_0 and set n to zero.
R/S	*02– 74*	Stop to enter number.
g x=0	*03– 25 6*	Is number zero?
g GTO 08	*04–25 7 08*	Yes, then go to line 08.
g CF_j	*05– 25 14*	No, store number ...
+	*06– 51*	and add it to total.
g GTO 02	*07–25 7 02*	Go back to line 02.
CLx	*08– 34*	Clear X-register to continue with program.

The above portion of the program will enable you to key in a series of numbers (after initializing CF_0) until you key in zero. Then the program will go to the next portion of the program. Let's look at the flowchart for the remaining portion of the program.



The n register has been counting the number of CFj entries, so we can use it in the decision-making process. Each time we recall a CFj entry, the contents of the n register are automatically decreased by 1.00. When n is decreased to zero, the program stops.

Now record the rest of your program:

Keystrokes	Display	
	08– 34	(Last recorded instruction.)
RCL n	*09– 22 11*	Recall n.
g x=0	*10– 25 6*	Compare to zero.
g GTO 00	*11–25 7 00*	If zero, go to line 00 and halt.
CLx	*12– 34*	If not, clear display.
RCL g CF_j	*13–22,25 14*	Recall CF_j entry.
g PSE	*14– 25 4*	Pause to display number.
g PSE	*15– 25 4*	(Two seconds!)
f %T	*16– 24 22*	Calculate % of total.
g PSE	*17– 25 4*	Pause to display % of total.
g PSE	*18– 25 4*	
CLx	*19– 34*	Clear display for next number.
g GTO 09	*20–25 7 09*	Back to line 09.

Let's discuss what you have recorded. Instructions in lines 09 through 20 tell the calculator to retrieve CF_j entries in the opposite order that you entered them. At the same time the calculator is told to display the number, then display the percentage that number is of the total sum. When the conditional test is YES, i.e., when n = 0, the program branches to line 00 and stops.

Why did we record all of the CLx instructions? In the first part of the program we calculated a sum. This sum is the base number for all of our %T calculations. In order to retain this sum in the Y-register, you must clear the display before a new number is keyed in, so that the base number is not moved up further into the automatic memory stack. If you press CLx before you key in a number, the contents of the Y-register are not disturbed. (Refer to RPN and the Automatic Memory Stack in *Your HP Financial Calculator*.)

How many numbers can we include in the sum? Press g MEM to find the number of registers currently allotted for storage.

Keystrokes	Display
g MEM	*P-22 r-18*

You can find the sum and %T of up to 18 numbers using this program.

Example: Find the percentage that each of the following numbers represent of the total sum of the numbers: \$54, \$14, \$83, \$144, and \$38. (Be sure to return to *run* mode.)

Keystrokes	Display	
g P/R		Back to *run* mode.
0 R/S	*0.00*	Initialize CF_0 with zero. Now key in the numbers.
54 R/S	*54.00*	
14 R/S	*68.00*	
83 R/S	*151.00*	
144 R/S	*295.00*	
38 R/S	*333.00*	Sum of all numbers entered.

Now press 0 R/S to tell the calculator that you have finished keying in the numbers so that it can branch to the %T portion of the program. The calculator will pause about 2 seconds to display each of the following:

Keystrokes	Display	
0 R/S	*38.00*	Last number you keyed in.
	11.41	% of total.
	144.00	Fourth number you keyed in.
	43.24	% of total.
	83.00	Third number.
	24.92	% of total.
	14.00	Second number.
	4.20	% of total.
	54.00	First number.
	16.22	% of total.
	0.00	Program stops.

To recall the total again, simply press x≷y.

Different programming approaches can be taken to solve the same problem. We could have keyed in all of the numbers manually, using [STO] or [CF$_j$], found their sum, and then using the latter part of the program, found the percent total of each. Or you may want to use [R/S] in place of two program lines of [g] [PSE], to allow enough time to write down the answers.

Hopefully, with the programming suggestions offered above and the editing techniques that follow, you can write and modify your own programs in a way that suits you best.

Section 8

Editing a Program

Even the most experienced programmer finds errors in his programs. These errors range from mistakes in the original formulas to mistakes in recording the program. Whenever they occur they need to be found and corrected, and your calculator is designed to make this error-checking process as easy as possible.

Finding the Error

One of the easiest ways to find out if your program is working properly is to work a test case in which you either know the answer or the answer can easily be determined. For example, if you have a program that calculates the sum of several numbers and then finds the percent each number is of the total sum, you can easily determine whether the program works by entering all like numbers (for example ten 10's) so that the percent total of each number is the same (10 is 10% of 100).

[SST] Execution

In longer programs a wrong test-case answer will seldom pinpoint a mistake. For these cases, you can slow down program execution by using [SST] *(single step)* in *run* mode. In *run* mode, [SST] will execute your program instructions one at a time. When you hold the [SST] key down in *run* mode, the program line number and keycode are displayed. When you release the [SST] key, the instruction is executed. Use [SST] on the program shown below to familiarize yourself with its operation.

Example Program: This program computes the net amount of an item when the sales tax is 7%. Toggle into *program* mode and clear program memory to display line 00. Then key in the list of keys shown below.

Keystrokes	Display	
[ENTER↑]	01-	31
7	02-	7
[%]	03-	23
[+]	04-	51

The program assumes that the cost of an item has been keyed into the X-register. To run the program, press [g] [P/R] . Now step through the program in slow motion using a value of 100 for the cost of the item.

Keystrokes	Display		
100	*100.*		
[g] [SST]	*01–*	*31*	When you hold [SST] down, the first instruction is displayed.
	100.00		When you release [SST] , the first instruction is executed.
[g] [SST]	*02–*	*7*	Again holding [SST] down displays the second instruction.
	7.		Again releasing [SST] executes the second instruction.
[g] [SST]	*03–*	*23*	Holding [SST] down displays the third instruction this time.
	7.00		And releasing [SST] executes the third instruction.
[g] [SST]	*04–*	*51*	And holding [SST] down again displays the fourth instruction.
	107.00		Releasing [SST] executes the fourth instruction.

You can see that it would be easy to spot a mistake in your program using the [SST] key.

When you hold the [BST] key down in *run* mode, the program line number and keycode for the previous line are displayed. When you release [BST] , the X-register is again displayed. If you press [R/S] in *run* mode after pressing [g] [BST] , the calculator will begin execution from the previous line in program memory. Now press [g] [BST] in *run* mode to review the program instructions of the above program.

Keystrokes	Display	
g BST	*04– 51*	Holding BST down in *run* mode displays the previous instruction.
	107.00	Releasing BST displays the original contents of the X-register.
g BST	*03– 23*	Holding BST down again displays the previous line in program memory.
	107.00	And releasing BST displays the original contents of the X-register again.

If you now toggle into *program* mode the third line will be displayed:

Keystrokes	Display
g P/R	*03– 23*

Cued Stops

If you have a program that is halted several times during execution for data entries, you may want to "identify" each stop by recording a familiar number into the program just before each R/S instruction. Then when the calculator stops execution because of the R/S instruction in the program, you can look at the displayed X-register to see the "identification number" for the required input. For example if your program contains eight stops for data inputs, it may be helpful to have the numbers 1 through 8 appear so you know which input is required each time. These identification numbers are helpful in editing a program.

Changing One Instruction

Changing or correcting one line of your program is easy with your HP-38E calculator because of the features built into it. Once the error has been found, use SST, BST, or GTO . in *program* mode to display the line *preceding* the line to be changed. For example, to change the instruction in line 06, you need to display line 05. If you wish to change the line, simply press the correct key or keys for line 06. They will write over and replace the incorrect information already stored in that line.

Example Program: The program represented below is designed to convert a given nominal interest rate to its continuous effective rate (see page 62). It assumes that 100 is stored in storage register R_0.

Keystrokes	Display
RCL 0	01- 22 0
÷	02- 71
g e^x	03- 25 22
1	04- 1
−	05- 41
RCL 0	06- 22 0
×	07- 61

(You could write the program using the same keystrokes you use manually: ENTER↑ 100 ÷ g e^x 1 − 100 ×.

Suppose that upon reviewing the program with the SST key, however, you discover that you have keyed in the following mistake-ridden program:

Keystrokes	Display	
RCL 0	01- 22 0	
÷	02- 71	
g LN	03- 25 23	Oops! You pressed the wrong key.
%	04- 23	And you pressed it again by mistake.
1	05- 1	
−	06- 41	
RCL 0	07- 22 0	
×	08- 61	

While the calculator is still in *program* mode, press g GTO . 02 or backstep to display line 02. Then correct the first mistake by keying in the correct keys for line 03.

Keystrokes	Display	
g GTO . 02	02- 71	First display this line.
g e^x	03- 25 22	Then press the correct keys for line 03.

With line 03 displayed you are ready now to correct line 04. Since this is an unwanted extra line you can use a [GTO] instruction to replace its contents rather than rewrite the whole program. Simply tell the calculator to go to the next line.

Keystrokes	Display	
	03- 25 22	Display line 03 to correct line 04.
[g] [GTO] 05	*04-25 7 05*	Use a [GTO] instruction to tell the calculator to go to the next desired line of the program.

Notice that we used [g] [GTO] [.] to *display* a program line in *program* mode and [g] [GTO] *(no decimal point)* to *record* a program line in *program* mode.

Now toggle into *run* mode by pressing [g] [P/R] . The example below will help you determine whether or not you have corrected the program.

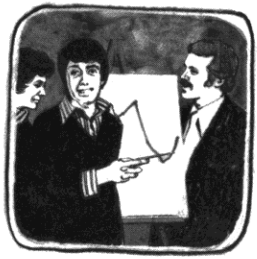
Example: If a savings institution quotes a nominal rate of 6¼%, compounded continuously, what is the effective rate? What is the effective rate of a 7% nominal interest rate, compounded continuously?

Keystrokes	Display
100 [STO] 0	*100.00*
6.25 [R/S]	*6.45*
7 [R/S]	*7.25*

Adding Instructions

If you have recorded a medium-sized program and have left out a crucial sequence of keystrokes right in the middle, you do not have to start over. The missing sequence of keystrokes can be recorded in the available lines following your program. You can then use the [GTO] key to make an unconditional branch to the sequence when it is needed and then make a second unconditional branch back to the main part of your program at the end of the sequence.

The sample inoperative program shown below should make this more clear. Four keys are missing between lines 03 and 04.



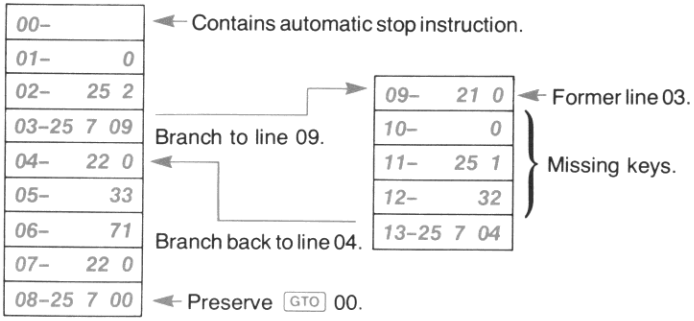
Example: This program, when corrected, will calculate the coefficients of the linear equation $y = A + Bx$ after you have entered a set of data points using [Σ+]. We want the y-intercept (A) to be calculated and displayed in the X-register, and the slope of the line (B) to be calculated and stored in the Y-register when the program has finished execution. Suppose you key in the following keystrokes in *program* mode.

Keystrokes	Display	
[g] [CL P]	00–	
0	01– 0	
[g] [ŷ,r]	02– 25 2	
[STO] 0	03– 21 0	
[RCL] 0	04– 22 0	← Missing three lines (0, [g] [x̂,r], and [CHS]) here.
[x≷y]	05– 33	
[÷]	06– 71	
[RCL] 0	07– 22 0	

When you correct this program, remember to preserve the automatically recorded [GTO] 00 instruction at the end of the program that tells the calculator when the program is done. With this in mind, branch to available program lines in program memory.

To do this, line 03 must be changed to [g] [GTO] 09, enabling you to form an unconditional branch to the line *after* the end of your program. Line 09 must contain the instruction formerly found in line 03. The missing keys are then stored in lines 10 through 12 and the instruction stored in line 13 is an unconditional branch back to line 04 in the main program.

The corrected program is shown below:



Multiple Programs

You can record more than one program in program memory, as long as you preserve the GTO 00 instruction at the end of each program. To run the first program, which starts at line 01, simply press R/S in *run* mode as you normally do. To run programs recorded later in program memory, address them with a GTO instruction before pressing R/S. For instance, if your second program begins on line 35, press 9 GTO 35 R/S in *run* mode to run the program.

Appendix A

Service and Maintenance

Your Hewlett-Packard Calculator

Your calculator is another example of the award-winning design, superior quality, and attention to detail in engineering and construction that have marked Hewlett-Packard electronic instruments for more than 30 years. Each Hewlett-Packard calculator is precision crafted by people who are dedicated to giving you the best possible product at any price.

AC Line Operation

Your calculator contains a rechargeable battery pack consisting of nickel-cadmium batteries. When you receive your calculator, the battery pack inside may be discharged, but you can operate the calculator immediately by using the ac adapter/recharger.

> **Note:** Do not attempt to operate the calculator from the ac line with the battery pack removed.

The procedure for using the ac adapter/recharger is as follows:

1. You need *not* turn the calculator off.
2. Insert the ac adapter/recharger plug into the connector on the top of the calculator, with the snap release tab on the plug facing toward the right side of the calculator.
3. Insert the power plug into a live ac power outlet.

> **CAUTION**
>
> The use of a charger other than the HP recharger supplied with the calculator may result in damage to your calculator.
>
> For the Continuous Memory model use only the "B" suffix version ac adapter/recharger shipped with your calculator (see product number on recharger). Earlier "A" suffix version rechargers will not damage your calculator, but may clear continuous memory when plugged in.

Battery Operation

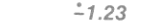
To operate the calculator from battery power alone, simply disconnect the recharger plug from the calculator by grasping the plug between your thumb and forefinger, squeezing to depress the snap release tab, and pulling gently. (Even when not connected to the calculator, the ac adapter/recharger may be left plugged into the ac outlet.)

Using the calculator on battery power gives the calculator full portability, allowing you to carry it nearly anywhere. A fully charged battery pack typically provides 3 hours of continuous operation. By turning the power off when the calculator is not in use, the charge on the battery pack should easily last throughout a normal working day.

Low Power

When you are operating from battery power and the batteries get low, a raised decimal is turned on at the far left of the display to warn you that you have between 1 minute and 25 minutes of operating time left.

If the display contains the low power indication, the minus sign looks like an incomplete divide sign.



To return to full power either connect the ac adapter/recharger to the calculator as described under AC Line Operation, or substitute a fully charged battery pack for the one in the calculator.

> **Note:** The ac adapter/recharger is a sealed unit and is not repairable. Return it to Hewlett-Packard if service is required.

Battery Charging

The rechargeable batteries in the battery pack are charged while you operate the calculator from the ac adapter/recharger. Batteries will charge with the calculator on or off, provided batteries are in place and recharger is connected. Normal charging times between the fully discharged state and the fully charged state are (depending on ac line voltage value):

Calculator off:	5 to 9 hours
Calculator on:	17 hours

Shorter charging periods will reduce the operating time you can expect from a single battery charge. Whether the calculator is off or on, the calculator battery pack is never in danger of becoming overcharged.

> **Note:** It is normal for the ac adapter/recharger to be warm to the touch when it is plugged into an ac outlet.

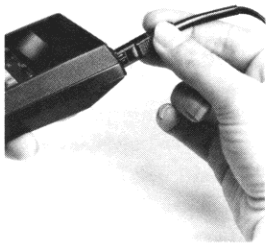
Battery Pack Replacement

If it becomes necessary to replace the battery pack, use only another Hewlett-Packard battery pack like the one shipped with your calculator.

> **CAUTION**
>
> Use of any batteries other than the Hewlett-Packard battery pack may result in damage to your calculator.

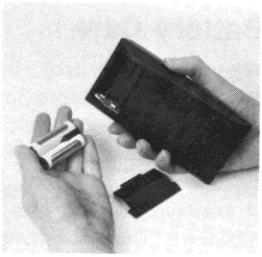
To replace the battery pack, use the following procedure:

1. Set calculator ON-OFF switch to OFF and disconnect the battery ac adapter/recharger from the calculator.



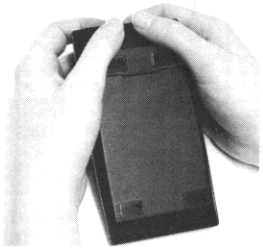
2. Press down on the short ridges of the battery door, close to the edge, until the door release snaps open. Slide the door open.

3. When door is removed, turn calculator over and gently shake, allowing the battery pack to fall into the palm of your hand.



4. Place the new battery pack into the calculator. Your calculator will only turn on if the battery pack is inserted correctly.

5. Insert battery door and slide door back into place.



6. Turn calculator over and turn power on to assure proper battery installation. If the display does not light, make sure the battery pack is correctly placed in calculator.

Battery Care

When not being used, the batteries in your calculator have a self-discharge rate of approximately 1 percent of available charge per day. After 30 days, a battery pack might have only 50 to 75 percent of its charge remaining, and the calculator might not even turn on. If a calculator fails to turn on, you should substitute a charged battery pack, if available, for the one in the calculator or plug in the ac adapter/recharger. The discharged battery pack should be charged for at least 12 hours.

If a battery pack will not hold a charge and seems to discharge very quickly in use, it may be defective. If the one-year warranty on the battery pack has not expired, return the defective pack to Hewlett-Packard according to the shipping instructions. (If you are in doubt about the cause of the problem, return the complete calculator along with its battery pack and ac adapter/recharger.) If the battery pack is out of warranty, see your nearest dealer to order a replacement.

WARNING

Do not attempt to incinerate or mutilate the battery pack—the pack may burst or release toxic materials.

Do not connect together or otherwise short-circuit the battery terminals—the pack may melt or cause serious burns.

Temperature Range

Temperature ranges for the calculator are:

Operating	0° to 45°C	32° to 113°F
Charging	15° to 40°C	59° to 104°F
Storage	−40° to 55°C	−40° to 131°F

Service

Blank Display

If the display blanks out, turn the calculator off, then on. If a display of numbers does not appear in the display check the following:

1. If the ac adapter/recharger is attached to the calculator, make sure it is plugged into an ac outlet.
2. Examine the battery pack to see if the contacts are dirty.
3. Substitute a fully charged battery pack, if available, for the one that was in the calculator.
4. If the display is still blank, try operating the calculator using the ac adapter/recharger (with the batteries in the calculator).
5. If, after step 4, the display is still blank, service is required. (Refer to Limited One-Year Warranty.)

Repair Policy

Hewlett-Packard calculators are normally repaired and reshipped within five (5) working days of receipt at any repair center. This is an average time and could possibly vary depending upon time of year and work load at the repair center.

Shipping Instructions

The calculator should be returned, along with completed Service Card, in its shipping case (or other protective package) to avoid in-transit damage. Such damage is not covered by warranty and Hewlett-Packard suggests that the customer insure shipments to the repair center. A calculator returned for repair should include the ac adapter/recharger and the battery pack. Send these items to the address shown on the Service Card. *Remember to include a sales slip or other proof of purchase with your unit.*

Whether the unit is under warranty or not, it is your responsibility to pay shipping charges for delivery to the Hewlett-Packard repair center.

After warranty repairs are completed, the repair center returns the unit with postage prepaid. On out-of-warranty repairs, the unit is returned C.O.D. (covering shipping costs and the service charge).

Limited One-Year Warranty

What We Will Do

The HP-38E and its accessories are warranted by Hewlett-Packard against defects in materials and workmanship for one year from date of original purchase. If you sell your calculator or give it as a gift, the warranty is automatically transferred to the new owner and remains in effect for the original one-year period. During the warranty period we will repair or, at our option, replace at no charge a product that proves to be defective provided that you return the product, shipping prepaid, to a Hewlett-Packard repair center.

How to Obtain Repair Service

Hewlett-Packard maintains repair centers in most major countries throughout the world. You may have your calculator repaired at a Hewlett-Packard repair center anytime it needs service, whether the unit is under warranty or not. There is a charge for repairs after the one-year warranty period. Please refer to the Shipping Instructions in this handbook.

The Hewlett-Packard United States Repair Center for handheld and portable printing calculators is located at Corvallis, Oregon. The mailing address is:

Hewlett-Packard
Corvallis Division • Service Department
1000 N.E. Circle Boulevard/P.O. Box 999
Corvallis, Oregon 97330

What Is Not Covered

This warranty does not apply if the product has been damaged by accident or misuse, or as a result of service or modification by other than an authorized Hewlett-Packard repair center.

No other expressed warranty is given. The repair or replacement of a product is your exclusive remedy. **ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS IS LIMITED TO THE ONE-YEAR DURATION OF THIS WRITTEN WARRANTY.** Some states do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you. **IN NO EVENT SHALL**

HEWLETT-PACKARD COMPANY BE LIABLE FOR CONSEQUENTIAL DAMAGES. Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Obligation to Make Changes

Products are sold on the basis of specifications applicable at the time of manufacture. Hewlett-Packard shall have no obligation to modify or update products once sold.

Warranty Information Number

If you have any questions concerning this warranty, please call toll-free (excpt in Alaska, Hawaii, and Oregon) 800/547-3400. (In Oregon call 757-2000.)

Programming and Applications Assistance

Should you need technical assistance concerning programming, calculator applications, etc., call Hewlett-Packard Customer Support at 503/757-2000. This is not a toll-free number, and we regret that we cannot accept collect calls. As an alternative, you may write to:

Hewlett-Packard
Corvallis Division/Customer Support
1000 N.E. Circle Boulevard
Corvallis, OR 97330

A great number of our users submit program applications or unique program key sequences to share with other HP owners. Hewlett-Packard will only consider using ideas given freely to us. Since it is the policy of Hewlett-Packard not to accept suggestions given in confidence, the following statement must be included with your submittal:

"I am voluntarily submitting this information to Hewlett-Packard Company. The information is not confidential and Hewlett-Packard may do whatever it wishes with the information without obligation to me or anyone else."

Further Information

Service contracts are not available. Calculator circuitry and design are proprietary to Hewlett-Packard, and service manuals are not available to customers. Should problems arise regarding repairs, please contact your nearest Hewlett-Packard repair center. The address for the United States Repair Center for handheld and portable printing calculators is:

Hewlett-Packard Company
Corvallis Division, Service Department
1000 N.E. Circle Boulevard/P.O. Box 999
Corvallis, Oregon 97330

Note: Not all Hewlett-Packard repair centers offer service for all models of HP calculators. However, you can be sure that service may be obtained in the country where you bought your calculator.

If you happen to be outside of the country where you bought your calculator, you can contact the local Hewlett-Packard repair center to see if service capability is available for your model. If service is unavailable, ship your calculator to the above address. A list of repair centers for other countries may be obtained by writing to the above address.

All shipping and reimportation arrangements are your responsibility.

Appendix B

More About IRR

Given a sequence of positive and negative cash flows, we hope that the calculator has enough information to determine whether an IRR answer exists, and what that answer is. And for the vast majority of cases, that's just what your HP-38E will do—find the unique IRR answer if it exists. But the IRR computation is so complex that if the cash flow sequence does not meet certain criteria, then sometimes the calculator is unable to determine whether or not an answer(s) exists.

Let's look at all of the possible outcomes to IRR as interpreted by your HP-38E:

1. *Error 7* . This indicates that there is no solution to IRR, probably because a mistake was made in data entry. You should review your cash flows to check for wrong values and sign changes.
2. **Positive answer.** If the calculator displays a positive IRR answer, then it is the *only* positive answer, but negative IRR answers may exist (although this is not probable).
3. **Negative answer.** If the calculator displays a negative IRR answer, then there *may* be more negative answers, as well as a unique positive IRR answer. If a unique positive answer exists, it is easy to find using the method described below.
4. *Error 3* . This error message indicates that the IRR problem may have multiple solutions, but the calculator doesn't have enough information to find an answer. The IRR problem does not fall into a category that the calculator can recognize. You may then help the calculator search for an IRR solution using the method described below.

Although the calculator will eventually reach one of the above outcomes, it may take a long time to get there. You may wish to terminate the IRR iterative process, by pressing any key, to see what interest rate the calculator has computed at that point. If you stop the iteration, you may continue searching for IRR as described below.

Searching for IRR. You can continue searching for IRR solutions, even after an *Error 3* indication, as follows:

1. Make a guess for the interest rate and key it in.
2. Press RCL g R/S.

Your guess will aid the calculator in its search, and if it finds an IRR answer near your guess, that answer will be displayed. Since the calculator cannot tell you the number of solutions that exist when there is more than one mathematically correct answer, you can continue to make guesses, pressing RCL g R/S after each one, to search for IRR solutions.

You can hasten this process by using the NPV function to help you make a good guess. Remember that a correct IRR solution will make the calculated NPV very small. So continue to guess interest rates and solve for NPV until the answer you obtain is reasonably close to zero. Then press RCL g R/S to calculate the IRR answer near your guess.

How would this work in case 3 above? The calculator displays a negative answer and you wish to check for a unique positive IRR. Key in successively larger guesses for i (starting from 0) and solve for NPV until you reach a sign change in your NPV outcomes. Then press RCL g R/S to find an IRR solution near the last interest rate obtained using the NPV key.

If you stop the IRR iterative process, you can test the interest obtained using NPV, and then restart the iteration by pressing RCL g R/S.

Appendix C

Error Messages

If you press any of the following function keys when the specified condition exists, you attempt an improper calculation. The calculator will display *Error* and a digit from 0 through 8, to aid you in determining the source of the error. Press any key to clear the error message from the display.

Error 0: Zero

[÷], when x = 0.
[1/x], when x = 0.
[√x], when x < 0.
[LN], when x ≤ 0.
[y^x] 1. When y = 0 and x ≤ 0.
2. When y < 0 and x is non-integer.
[Δ%], when y = 0.
[%T], when y = 0.
[STO] [÷] 0 through 6, when x = 0.
[n!] 1. When x is non-integer.
2. When x < 0.

Error 1: Storage Register Overflow

[12×]
[STO] [+] 0 through 6
[STO] [÷] 0 through 6
[STO] [×] 0 through 6
[STO] [−] 0 through 6
[Σ+]
[Σ−] } HP-38E only.

} Magnitude of number in register greater than $9.999999999 \times 10^{99}$.

Error 2: Statistics

Improper data stored in statistical registers R_1 through R_6.

[x̄], when n = 0.
[x̄w], when Σx = 0.
[s] 1. When n = 0 or n = 1.
2. When $SS_x < 0$.
3. When $SS_y < 0$.
$\hat{y}$ 1. When n = 0.
2. When $n(\Sigma x^2) - (\Sigma x)^2 = 0$.
$\hat{x}$ 1. When n = 0.
2. When $n(\Sigma y^2) - (\Sigma y)^2 = 0$.
r, when $[n(\Sigma x^2) - (\Sigma x)^2] \times [n\Sigma y^2) - (\Sigma y)^2] \leq 0$.

Y-register not cleared to zero before beginning accumulations of a one-variable (*x*) series.

Error 3: Amortization or IRR

[AMORT] 1. When x ≤ 0.
2. When x is non-integer.
[IRR] Not enough information to solve. May have multiple solutions. Help the calculator search: key in a guess and press [RCL] [9] [R/S].

Error 4: Improper Memory Address

1. Attempting to enter more than 99 lines of programming.
2. Attempting to execute a [GTO] into registers not defined as program lines. Press [9] [MEM] to display current memory allocation.
3. Attempting storage register arithmetic in storage registers R_7 through R_9, $R._0$ through $R._9$. Storage registers R_0 through R_6 are allocated for storage register arithmetic.

Error 5: Compound Interest

Bad input to top row functions.

[n] 1. When PMT $\leq -PV \times i$, then the principal does not decrease.
2. When PMT = FV × i, then n is undefined.
3. When the values in i, PV, PMT, FV are such that no solution exists for n.

[i] 1. When PMT = 0 and $n < 0$.
2. When PMT $\neq 0$ and $n < 0.995$.
3. When cash flows are illegal.

[n], [PV], [PMT], [FV]; when [i] ≤ -100.

Error 6: Discounted Cash Flow Analysis

[NPV] 1. When [n] > 20 or [n] $>$ r as defined by [MEM].
2. When [n] < 0.
3. When [n] is non-integer.
4. When [i] ≤ -100.

[IRR] 1. When [n] > 20 or [n] $>$ r as defined by [MEM].
2. When [n] < 0.
3. When [n] is non-integer.

[CF$_j$] 1. When out of range on [n].
2. Referring to storage register already defined as program lines.

[N$_j$], when $x > 99$, $x < 0$, or x noninteger.

Error 7: IRR

[IRR] 1. When cash flows all of same sign.
2. When no IRR solution exists.

Error 8: Calendar

[△DAYS], illegal date, or improper date format; check switch.

[DATE] 1. Improper date format.
2. Attempting to add days beyond calculator's date capacity.

Error 9: Failed Self-Check
([STO] [ENTER↑])

Pr Error (HP-38C only)
Continuous Memory cleared because of power failure.

Appendix D

Continuous Memory (HP-38C only)

When you turn off your calculator, the following information is retained:

- Contents of the financial registers and the storage registers.
- All programs that are loaded into the calculator.

Numbers in the stack and LAST X register are not preserved when you turn off the calculator. Regardless of where you stopped in a program, the calculator returns to line 00 (top of program memory) when you turn it on again.

Executing the self-check does not clear program memory, storage registers, or financial registers.

Continuous Memory requires that the batteries be kept in the calculator. If the low power indicator appears in the display, turn your calculator off immediately and connect it to an ac outlet or insert a new battery pack. Continuous Memory requires that batteries be replaced as quickly as possible. Normally you have a minimum of 5 seconds to change the batteries. Leaving the batteries out of the calculator for extended periods, or allowing the batteries to discharge completely will result in loss of the information in Continuous Memory.

If you drop or traumatize your calculator, or if power to the Continuous Memory is interrupted whether the calculator is off or on, the contents of program memory and the data storage registers may be lost. If this occurs, when the calculator is then turned on, the display will show Pr Error. To restore the display, ensure that the battery is charged, or connect the ac adapter/recharger, and press any key.

CAUTION

Switching off the HP-38C during execution of either a program or a function may clear or alter instructions in program memory and data in the financial or storage registers.

Index

A

AC line operation, **126**
Accumulated interest, **32-34, 45, 46, 47**
Active keys, **10-11, 81**
Addition, **19**
Allocation, automatic memory, **85-88**
Alteration, number, **63-65**
Amortization, **28, 29, 45-48**
  Accumulated interest, **46**
  Remaining balance, **46**
Annual interest rate, see Interest rate
Antilogarithms, **61**
Applications assistance, **133**
Arithmetic, **19**
Arithmetic average, see Mean
Assistance, programming and applications, **133**
Automatic memory allocation, **85-88**
Automatic memory stack, **6**
Automatic run mode, **81**
Automatic stop instruction, **82**
Average, see Mean

B

Back step, **93, 119-120**
Balance, remaining, **46**
Balloon payment, **36, 63-64**
Battery care, **130**
Battery charging, **127**
Battery, duration of power, **99**
Battery operation, **127**
Battery pack replacement, **128-129**
Blank display, **131**
Blue prefix key, **18**

Branching, conditional, **108-117**
Branching, unconditional, **106-108**

C

Calendar, **24-26**
  Date format, **24**
  Day of the week, **26**
  Days between dates, **24**
  Future or past dates, **25**
Cash flow diagram, **30**
Cash flow, discounted, **28, 48-49**
  Initial (CF_0), **28, 49**
  Subsequent (CF_j), **28, 49**
  Number of times a cash flow occurs (N_j), **28, 49**
Cash flow sign convention, **31, 45, 49, 54**
Cash flow zero, **49**
Cash flow, discounted, **28, 48-49**
  Group, example, **52, 55, 57**
  Individual, example, **49, 56**
Changing cash flow entries, **56**
Changing one program instruction, **120-122**
Charging the batteries, **127**
Clearing a program, **82, 84**
Clearing an error, **100, 138**
Clearing financial registers, **29**
Clearing prefix keys, **18**
Clearing statistical registers, **8, 67**
Clearing storage registers, **7**
Clearing the display (X-register), **6**
Coefficients of linear equation, **75**
Compound interest calculations, **31-32, 34-48**
  Solving for future value, **43-45**
  Solving for present value, **39-41**
  Solving for the number of compounding or payment periods, **34-37**
  Solving for the periodic interest rate, **37-39**
  Solving for the periodic payment amount, **42-43**
Compounding periods, example, **36-37**
Conditionals, **109**
Conditional test, **110**

Conditional tests and conditional branches, **108-117**
Continuous effective rate, **62**
Continuous Memory, **140**
Correcting statistical data, **67**
Correcting cash flows for NPV and IRR, **56-57**
Correlation coefficient, **72-73**
Cued stops, **120**

D

Date format, **24**
Dates, past or future, **25**
Day of the week, **26**
Days between dates, **24**
Decision-making, **104-117**
Deleting and correcting statistical data, **67**
Deleting program instructions, **121-122**
Deviation, standard, **69**
Discounted cash flow analysis, **48-57**
  Cash flow zero (CF_0), **28, 49**
  Changing cash flow entries, **56**
  Entering cash flows, **49, 54**
  Functions, **28, 29**
  Internal rate of return (IRR), **51, 53-56, 136-137**
  Net present value (NPV), **49-53, 55**
  Number of times a cash flow occurs (N_j), **28, 49**
  Reviewing cash flow entries, **56**
  Subsequent cash flows (CF_j), **28, 49**
Display, error, **138-139**
Display, low power, **127**
Display financial values, **29**
Division, **19**
DO if TRUE rule, **110**

E

Editing a program, **118-124**
Editing cash flow entries, **56-57**
Effective rate, **61, 62**
End of a program, **84-85, 123-124**
Entering cash flows, **49-54**

Equation, linear, **75**
Error conditions, **138-139**
Error display, **138**
Error messages, **138-139**
Error stops, **100**
Executing instructions, **81, 84, 85, 118**
Exponential functions, **61-62**
Exponentiation: raising a number to a power, **59-60**
Exponents of ten, keying in, **23**
Extracting roots, **60**

F

Factorials, **62**
Financial functions, **28-57**
Financial interest calculations, **30-57**
Financial registers, **28**
  Clearing financial registers, **29**
  Displaying financial values, **29**
  Reentering values, **29**
Financial values, **31**
Flowcharts, **102-106**
Fractional portion of a number, **65**
Function and key index, **6-9**
Functions, financial, **28-57**
Functions, mathematical, **58-65**
Functions, statistical, **66-77**
Future or past dates, **25**
Future value, **28, 32, 43-45**

G

Going to a line number, **93-94**
  Branching execution, **106-108**
  Line 00, **84**
  Multiple programs, **124**
Gold prefix key, **18**
Group cash flows, **52, 55, 57**

I

Improper operations, see Error conditions
Individual cash flows, **49, 56**
Initial cash flow, (CF_0), **28, 29, 49, 54**
Instructions, deleting, **121-122**
Instructions, executing, **81, 84, 85, 118**
Integer portion of a number, **64**
Interest, accumulated, **32-34, 45, 46, 47**
Interest rate, **28, 31, 37-39**
Interest rate, annual, **37-39**
Interest, compound, **34-45**
Interest, simple, **32-34**
Internal rate of return (IRR), **28, 51, 53-56, 136-137**
Interrupting a program, **94-100**

K

Key index, programming, **10**
Keyboard, **6, 18**
Keyboard index, function, **6**
Keycodes, **82-83**
Keying in exponents of 10, **23**
Keys, prefix, **18**

L

LAST X register, **22, 64, 65**
Limited one-year warranty, **132**
Line 00, **82**
Linear coefficients program, **123-124**
Linear equation, coefficients of, **75**
Linear estimates, **74-77**
Linear regression, **72-77**
  Coefficients of linear equation, **75**
  Correlation coefficient, **72-73**
  Linear equation, **75**
  Linear estimates, **74-77**
  Regression line, **75**
  Slope of a line, **73, 75**

Logarithms, **61**
Looping, **106-108**
Low power display, **127**

M

Mantissa function, **22**
Manual problem solving, **12**
Manual run mode, **81**
Margin, **20**
Markup, **20**
Mathematical functions, **58-65**
Mean, **67-69**
Mean, weighted, **70-71**
Memory, automatic allocation, **85-88**
Memory, map, **6, 87**
Memory map, function (MEM), **50, 87-88**
Merged keycodes, **84, 92**
Mistakes, recovering from, **22**
More about IRR, **136-137**
Multiple programs, **124**
Multiplication, **19**

N

Natural logarithms, **61**
Negative cash flows, **31**
Net present value (NPV), **28, 49-53, 55**
Nominal rate, **61**
Nonrecordable operations, **81**
Number alteration keys, **63-65**
Number, fractional portion of, **65**
Number, integer portion of, **64**
Number, rounding a, **63-64**
Number, recovering a, **22**
Number of periods, **28, 31, 34-37**
Number of times a cash flow occurs (N_j), **28, 49**
Numbers, raising to powers, **59**
Numbers, recalling, **7**
Numbers, storing, **6**

O

Operating temperatures, **130**
Operation, battery, **127**
Operations, improper, see Error conditions
Overflow calculations, **100**

P

Past or future dates, **25**
Pausing to view output, **94, 97-99**
Payment amount, **28, 31, 32, 42-43**
Payment periods, **28, 31, 34-37**
Payment switch, **24, 29, 31**
Percent difference between two numbers, **20**
Percent of total, **21**
Percentages, **19**
Periodic interest rate, **37-39**
Periodic payment amount, **42-43**
Periods, compounding or payment, **28, 31, 34-37**
Periodic payment amount, see Payment amount
Powers, raising numbers to, **59-60**
Prefix keys, **18**
Present value, **28, 32, 39-41**
Principal, **40-41, 45-48**
Program, **12, 80**
Program changes, verifying, **93**
Program, clearing, **82, 84**
Program, editing, **118-124**
Program, ending, **84-85, 123-124**
Program interruptions, **94-100**
Program, introductory, **82**
Program memory, **6, 84, 86, 87, 88**
Program mode, **81**
Program/run, **13, 81**
Program running, **84**
Program, single-step execution, **118-119**
Program, single-step viewing without execution, **93**
Program stops, **99-100**
Programmed problem solving, **13**

Programming assistance, **133**
Programming key index, **10-11**

R

Raising a number to a power, **59**
Range, temperature, **130**
Recalling financial values, **29**
Recalling numbers from memory, **7**
Recharging procedure, **127**
Reciprocals, **58**
Reentering financial values, **29**
Registers, map, **6**
Regression line, **75**
Remaining balance, **46**
Repair centers, **132, 134**
Repair policy, **131**
Replacing battery pack, **128-129**
Resetting program memory, **82, 84**
Reviewing cash flow entries, **56-57**
Roots, **60**
Rounding a number, **63-64**
Rounding payments, **46**
Run/stop, **85, 94**
Running a program, **14, 81, 84**

S

Schedule, amortization, **45-48**
Service, **131-134**
Shipping instructions, **131**
Sign, in cash flows, **31**
Simple interest, **28, 32**
  Thirty-day month basis, **32**
  Actual-day basis, **34**
Single step, viewing without execution, **93**
Single step execution, **118-119**
Slope of a line, **73, 75**
Square root, **58-59**
Stack, automatic memory, **6**

Standard deviation, **69**
Statistical functions, **66**
Statistical registers, **66-67**
  Clearing, **8**
  Deleting and correcting data, **67**
  Viewing, **67**
Stopping a program, **94**
Storage registers, **6, 86-87**
  Arithmetic, **6**
  Clearing, **7**
  Recalling, **7**
  Storing, **6**
Storage registers, **86, 87, 88**
Storing numbers, **6**
Subsequent cash flows (CF_j), **28**
Subtraction, **19**
Summations, **66**

T–Z

Temperature range, **130**
Transfer of warranty, **132**
Unconditional branching and looping, **106**
Viewing the statistical registers, **67**
Warranty information toll-free number, **133**
Warranty, limited one-year, **132**
Weighted mean, **70**
Yield, **39, 40, 51, 53**

Information

The warranty period for your calculator and/or accessory is one year from date of purchase. Hewlett-Packard will assume that any unit returned without a copy of proof of purchase (sales slip or validation) is out of warranty. Should service be required, please return your calculator, charger, batteries and this card protectively packaged to avoid in-transit damage. Such damage is not covered under warranty.

Inside the U.S.A.

Return items safely packaged directly to:

Hewlett-Packard
Corvallis Division, Service Department
1000 N.E. Circle Boulevard/P.O. Box 999
Corvallis, Oregon 97330

We advise that you insure your calculator and use priority (AIR) mail for distances greater than 300 miles to minimize transit times. All units will be returned by fastest practical means.

Outside the U.S.A.

Where required please fill in the validation below and return your unit to the nearest designated Hewlett-Packard Sales and Service Office. Your warranty will be considered invalid if this completed card is not returned with the calculator.

Model No. _____ Serial No. _____

Date Received _____

Invoice No./Delivery Note No. _____

Sold by:

Service Card

This card must be completed and returned with your calculator and/or recharger, and batteries. Return of this card is considered authorization for Hewlett-Packard to make all repairs necessary to return the calculator to normal working order and to charge the cost of those repairs to the owner for units out of warranty.

Owner's Name _____ Date _____

Street Address _____

City _____ State _____ Zip Code _____

Home Phone _____ Work Phone _____

Date Purchased _____

What Is The Problem Area?

- ☐ Intermittent Problem
- ☐ Printer (Enclose sample)
- ☐ Keyboard
- ☐ Programming
- ☐ Display
- ☐ Recharger/Battery
- ☐ Prerecorded Program/Reader

Describe Problem: _____

Model No. _____ Serial No. _____

Preferred method of payment for out-of-warranty repairs.
If not specified, unit will be returned C.O.D.

☐ VISA ☐ Master Charge

Card No. _____ Expiration Date _____

Name appearing on credit card _____

☐ Purchase Order. Companies with established Hewlett-Packard credit only. (Include copy of purchase order with shipment.)

P.O. Number _____

Authorized Signature _____

Calculator Catalog and Buying Guide Request Card

Thank you for your order.

A friend or associate might also want to know about Hewlett-Packard calculators. If you would like us to send the current issue of the *Hewlett-Packard Personal Calculator Digest* (The HP Magazine and Product Catalog), please write his/her name and address on this postage-paid Request Card.

Primary Interest:

- ☐ Scientific Calculators
- ☐ Fully-Programmable Calculators
- ☐ Business Calculators
- ☐ All

Name _____

Title _____

Company _____

Street _____

City _____ State _____ Zip _____

Valid in U.S. only

430N

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 CORVALLIS, OREGON

Postage will be paid by

Hewlett-Packard
1000 N.E. Circle Blvd.
Corvallis, Oregon 97330

HEWLETT
PACKARD

1000 N.E. Circle Blvd., Corvallis, OR. 97330

For additional sales and service information contact your local Hewlett-Packard Sales Office or call 800/547-3400. Excludes Alaska and Hawaii. (In Oregon call 758-1010.)

00038-90038 Rev. B 8/80

Printed in U.S.A.

Scan Copyright ©
The Museum of HP Calculators
www.hpmuseum.org

Original content used with permission.

Thank you for supporting the Museum of HP Calculators by purchasing this Scan!

Please to not make copies of this scan or make it available on file sharing services.