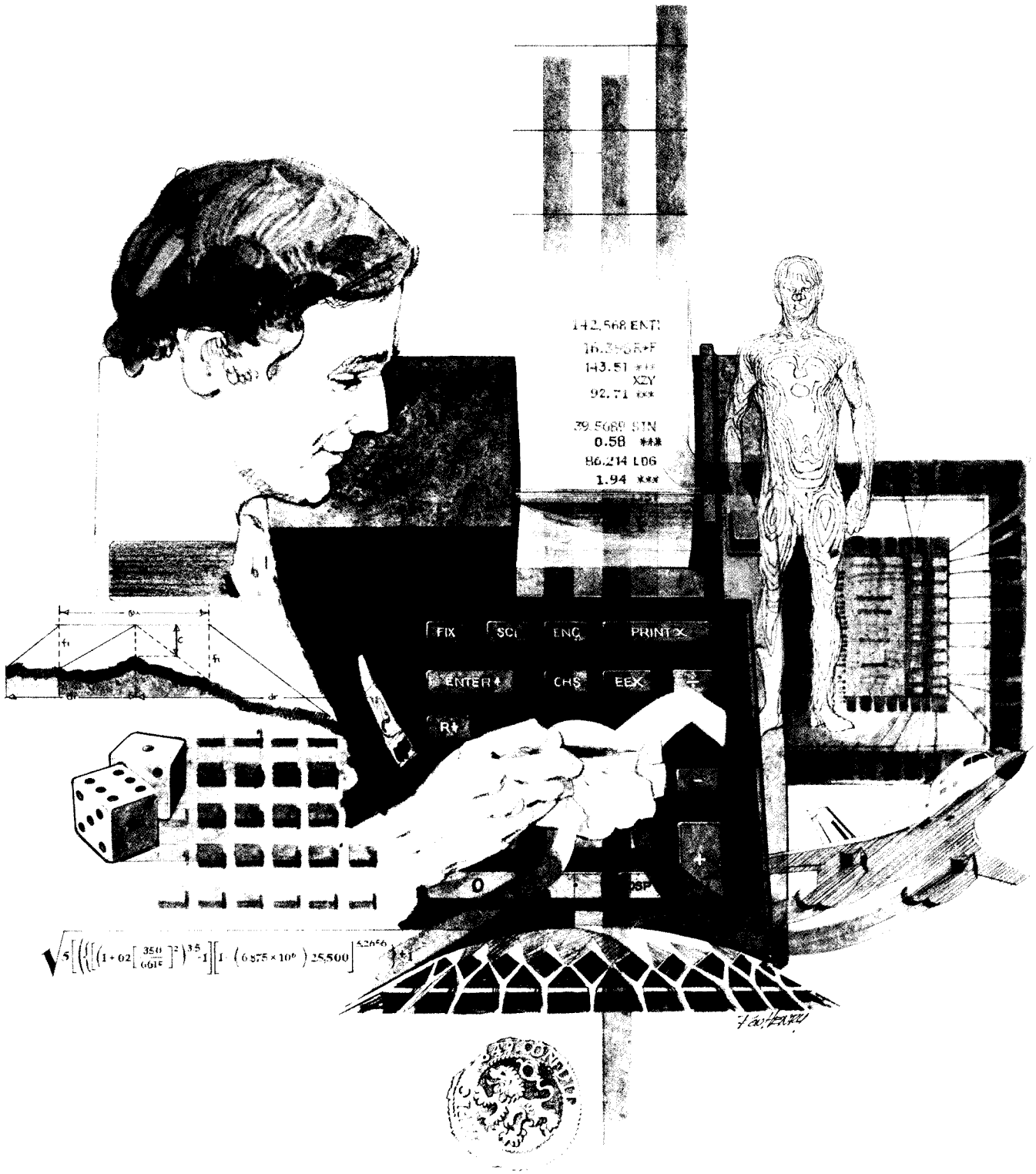
HEWLETT-PACKARD

HP-67/HP-97

Users' Library Solutions

Home Management



INTRODUCTION

In an effort to provide continued value to it's customers, Hewlett-Packard is introducing a unique service for the HP fully programmable calculator user. This service is designed to save you time and programming effort. As users are aware, Programmable Calculators are capable of delivering tremendous problem solving potential in terms of power and flexibility, but the real genie in the bottle is program solutions. HP's introduction of the first handheld programmable calculator in 1974 immediately led to a request for program **solutions** — hence the beginning of the HP-65 Users' Library. In order to save HP calculator customers time, users wrote their own programs and sent them to the Library for the benefit of other program users. In a short period of time over 5,000 programs were accepted and made available. This overwhelming response indicated the value of the program library and a Users' Library was then established for the HP-67/97 users.

To extend the value of the Users' Library, Hewlett-Packard is introducing a unique service—a service designed to save you time and money. The Users' Library has collected the best programs in the most popular categories from the HP-67/97 and HP-65 Libraries. These programs have been packaged into a series of low-cost books, resulting in substantial savings for our valued HP-67/97 users.

We feel this new software service will extend the capabilities of our programmable calculators and provide a great benefit to our HP-67/97 users.

A WORD ABOUT PROGRAM USAGE

Each program contained herein is reproduced on the standard forms used by the Users' Library. Magnetic cards are not included. The Program Description I page gives a basic description of the program. The Program Description II page provides a sample problem and the keystrokes used to solve it. The User Instructions page contains a description of the keystrokes used to solve problems in general and the options which are available to the user. The Program Listing I and Program Listing II pages list the program steps necessary to operate the calculator. The comments, listed next to the steps, describe the reason for a step or group of steps. Other pertinent information about data register contents, uses of labels and flags and the initial calculator status mode is also found on these pages. Following the directions in your HP-67 or HP-97 **Owners' Handbook and Programming Guide,** "Loading a Program" (page 134, HP-67; page 119, HP-97), key in the program from the Program Listing I and Program Listing II pages. A number at the top of the Program Listing indicates on which calculator the program was written (HP-67 or HP-97). If the calculator indicated differs from the calculator you will be using, consult Appendix E of your **Owner's Handbook** for the corresponding keycodes and keystrokes converting HP-67 to HP-97 keycodes and vice versa. No program conversion is necessary. The HP-67 and HP-97 are totally compatible, but some differences do occur in the keycodes used to represent some of the functions.

A program loaded into the HP-67 or HP-97 is not permanent—once the calculator is turned off, the program will not be retained. You can, however, permanently save any program by recording it on a blank magnetic card, several of which were provided in the Standard Pac that was shipped with your calculator. Consult your **Owner's Handbook** for full instructions. A few points to remember:

The Set Status section indicates the status of flags, angular mode, and display setting. After keying in your program, review the status section and set the conditions as indicated before using or permanently recording the program.

REMEMBER! To save the program permanently, **clip** the corners of the magnetic card once you have recorded the program. This simple step will protect the magnetic card and keep the program from being inadvertently erased.

As a part of HP's continuing effort to provide value to our customers, we hope you will enjoy our newest concept.

TABLE OF CONTENTS

INCOME TAX PLANNING - I 1
Compares taxes owed when based on itemized deduction, standard deduction, or income averaging.

TRUE COST OF INSURANCE POLICY 8
Computes the true cost per $1,000 of insurance protection and rate of return on cash-value in an insurance policy.

AUTOMOBILE COST/TIRE COST COMPARISON 12
Two programs on one card to aid in evaluating automobile purchases: 1 - Compares the cost of ownership of two cars; 2 - Compares up to three different tires to determine the best buy based on miles expected per dollar spent.

COMPARISON SHOPPING 19
Computes the unit price of any number of items and retains the lowest priced item for display.

TIME & CHARGES RUNNING TOTAL 23
Shows time and charges for telephone calls while you talk.

RECONCILE CHECKING ACCOUNT 27
Aid in reconciling personal checkbook tallies with the bank statement.

SAVINGS ACCOUNT COMPOUNDED DAILY 31
Uses a calendar program to calculate precise interest.

ACCUMULATED INTEREST/REMAINING BALANCE 35
Computes total interest paid and remaining balance and also generates a period by period amortization schedule.

STOCK PORTFOLIO VALUATION AND DATA CARD 42
Evaluates a portfolio of stocks given the current market price per share and the annual dividend. The output includes the new portfolio value, the percent change in value, and the current dividend yield as a percent of the current market value. The program uses data cards produced by the Portfolio Data Card program.

TRUE ANNUAL GROWTH RATE OF AN INVESTMENT PORTFOLIO 54
Considers the time value of money in determining a rate of return of an active portfolio; a portfolio in which funds are being invested and withdrawn at various times.

DIET PLANNING 65
Estimates the required calories/day for an individual to maintain, gain, or lose weight based on activity levels and estimated basal metabolism.

Program Description I

Program Title Income Tax Planning - I

Contributor's Name Richard D. Rutter
Address Arthur Young & Co. 780 N. Water St.
City Milwaukee **State** Wi **Zip Code** 53202

Program Description, Equations, Variables This program calculates regular, alternative, and and average income taxes for individuals using IRS forms 1040, schedule D, and schedule G. Although the program was originally written prior to the Tax Reducation and Simplification Act of 1977 (which effects 1977 returns) the changes in the law have only had a minor effect on the program results. The tax amount computed for ordinary income differs slightly from that arrived at through the use of Table A through D (adjusted incomes less than $20,000 ($40,000 for joint returns)) but the differences are well within the tolerances required for tax planning. For filing purposes, the new tables (A through D) should be used for calculating ordinary income wherever specified by the IRS. If schedule x, y, and z are specified for tax computation, however, the program answers are exact.

The following mnemonics are in the accompanying documentation

OTI = Form 1040 Line 34
- (exemptions x 750)
- 3200 Joint
or
1600 Separate
or
2200 Single
-(.5 x C G)

C G = Capital Gains (Schedule D, line 13)

4 yr TI = Total Taxable income for preceding 4 years (see next page)

Computed tax amounts do not include the income tax credit. For the exact net tax amount, use the program Tax Computation Schedule and the tax on ordinary income generated by this program.

Program Description I

Program Title

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables

4 Yr TI - The increase in personal exemption for the current year (1977) has resulted in an increase in the 4 year base period total as implemented in the new income averaging schedule G. The effect is an increase in the averaged tax of approximately .5%. For planning purposes this is not a significant amount. However, if the exact income averaged tax is desired, add the following amounts to the 4 year taxable income.

+ $2133 Joint

or

+ $1067 Separate

or

+ $1467 Single

Operating Limits and Warnings

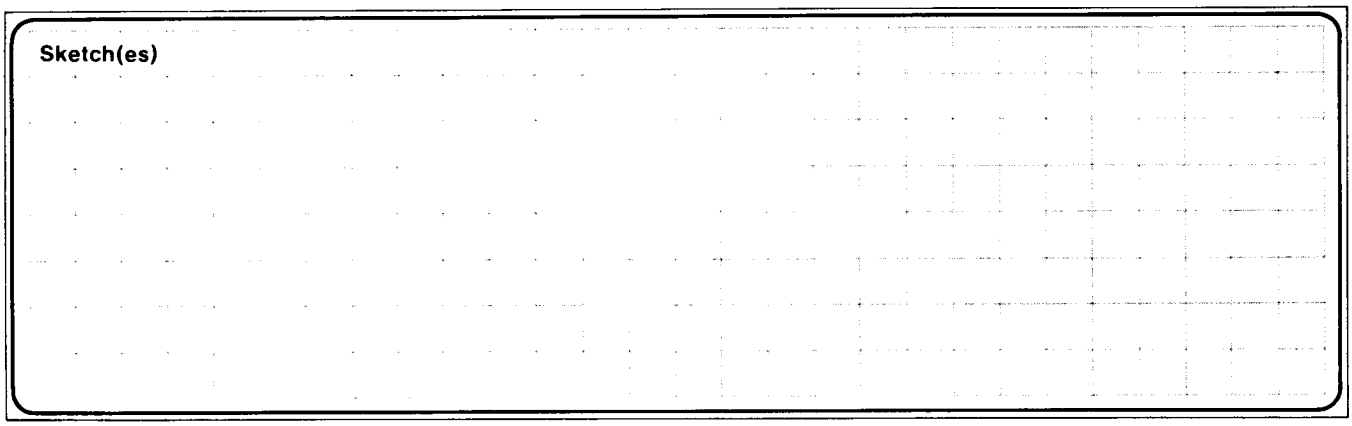
Tax calculations cannot be performed for values less than $1,000 (if attempted, error code '9' will flash in the display). All input data must conform with the following limits: OTI $\geq$ $1,000

CG > 0

4 yr TI > 0

Program Description II

Sketch(es)



Sample Problem(s)

Mr. and Mrs. Smith will file a joint return claiming 2 exemptions for tax year 1977. The following data are applicable.

1040 Line 34	100,000
Capital Gains (CG)	10,000
4 Year TI	160,000

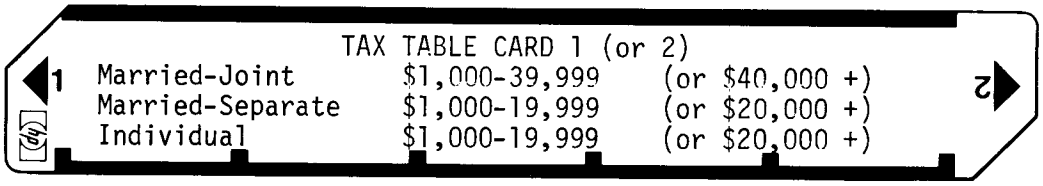
2.00	GSBH	
100000.00	ENT↑	line 34
750.00	ENT↑	
2.00	×	
1500.00	***	net exemption
	-	
98500.00	***	
3200.00	-	standard deduction
95300.00	***	
		capital gains
10000.00	STOC	
2.00	÷	
	-	-1/2 C G
90300.00	***	
	STOB	OTI
160000.00	STOD	4 yr TI
	GSBE	
3.00	T	
42360.00	Z	regular
41860.00	Y	alternative
40529.00	X	averaged

Reference(s)

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Load program card sides 1 and 2			
2.	Start by specifying tax table card 1 or 2 to be loaded. Be sure to select the pair of tax table cards to correspond to the type of tax return (E.G. joint return). '1' or '2' will flash in the display until the specified tax table card has been loaded. During execution the program will automatically request tax table card loading (if necessary) by flashing '1' or '2' in the display until the required tax table card has been loaded.	1 or 2	A	1 or 2
3.	Store the following data:	OTI	STO B	OTI
		C G	STO C	C G
		4 Yr TI	STO D	4 YR TI
4.	Run the program to calculate regular, alternative, and averaged taxes. Results are left in the stack and may be reviewed by:		E	'3.00' 'Reg Tax' 'Alt Tax' 'Ave Tax'
			g STK	
	NOTE: Execution times range from 3-45 seconds.			
5.	Tax calculations must be for taxable amounts above $1,000. If a tax calculation for a lesser amount is attempted. The program will halt with error code '9' in the display.			

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Record the following data on pairs of data cards for use with Income Tax Planning I and II programs. Record only the Tax Table(s) that you will use.			

REG	MARRIED-JOINT	MARRIED-SEPARATE	INDIVIDUAL
	CARD 1		
R0	1.0401	1.0201	1.0201
R1	1.0001415	0.	0.
R2	2.0002916	0.	0.
R3	3.0004517	1.0001416	1.0001416
R4	4.0006219	2.0003119	2.0003119
R5	8.0013822	4.0006922	4.0006921
R6	12.0022625	6.0011325	6.0011124
R7	16.0032628	8.0016328	8.0015925
R8	20.0043832	10.0021932	10.0020927
R9	24.0056636	12.0028336	12.0026329
R 0	28.0071039	14.0035539	14.0032131
R 1	32.0086642	16.0043342	16.0038334
R 2	36.0103445	18.0051745	18.0045136
R 3	52000.	26000.	38000.
	CARD 2		
R0	2.0402	2.0202	2.0202
R1	40.0121448	20.0060748	20.0052338
R2	44.0140650	22.0070350	22.0059940
R3	52.0180653	26.0090353	26.0075945
R4	64.0244255	32.0122155	32.0102950
R5	76.0310258	38.0155158	38.0132955
R6	88.0379860	44.0189960	44.0165960
R7	100.0451862	50.0225962	50.0201962
R8	120.0575864	60.0287964	60.0263964
R9	140.0703866	70.0351966	70.0327966
R 0	160.0835868	80.0417968	80.0393968
R 1	180.0971869	90.0485969	90.0461969
R 2	200.1109870	100.0554970	100.0530970
R 3	52000.	26000.	38000.

KEYS	
STO	0
STO	1
STO	2
STO	3
STO	4
STO	5
STO	6
STO	7
STO	8
STO	9
P S	
STO	0
STO	1
STO	2
STO	3
P S	W/DATA
STO	0
STO	1
STO	2
STO	3
STO	4
STO	5
STO	6
STO	7
STO	8
STO	9
P S	
STO	1
STO	2
STO	3
P S	W/DATA

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBL0	21 00	Tax Calculation Subroutine
002	EEX	-23	
003	3	03	
004	÷	-24	
005	STOE	35 15	
006	1	01	
007	X≤Y?	16-35	
008	GTOE	22 06	
009	9	09	
010	R/S	51	
011	*LBL6	21 06	
012	RCLI	36 46	
013	FRC	16 44	
014	EEX	-23	
015	3	03	
016	×	-35	
017	ENT↑	-21	
018	INT	16 34	
019	RCLE	36 15	
020	X≠Y	-41	
021	X>Y?	16-34	
022	GTO1	22 01	
023	2	02	
024	GTOB	22 12	
025	*LBL1	21 01	
026	1	01	
027	*LBLB	21 12	
028	R↑	16-31	
029	FRC	16 44	
030	EEX	-23	
031	1	01	
032	×	-35	
033	INT	16 34	
034	X=Y?	16-33	
035	GTO5	22 05	
036	R↓	-31	
037	GTO3	22 03	
038	*LBLA	21 11	Entry for initial Tax Table load
039	SF2	16 21 02	
040	*LBL3	21 03	Tax Table Load Routine
041	RCL0	36 00	
042	ABS	16 31	
043	R↓	-31	
044	1	01	
045	3	03	
046	STOI	35 46	
047	X≠Y	-41	
048	*LBL2	21 02	
049	MRG	16-62	
050	PSE	16 51	
051	F3?	16 23 03	
052	GTO4	22 04	
053	GTO2	22 02	
054	*LBL4	21 04	
055	RCL0	36 00	
056	STOI	35 46	
057	LSTX	16-63	
058	STO0	35 00	
059	R↓	-31	
060	INT	16 34	
061	X=Y?	16-33	
062	GTO9	22 09	
063	R↓	-31	
064	GTO3	22 03	
065	*LBL9	21 09	
066	F2?	16 23 02	Exit for initial Tax Table Load
067	RTN	24	
068	*LBL5	21 05	Tax Table Search Routine
069	RCLI	36 46	
070	FRC	16 44	
071	1	01	Perform indirect read loop through Tax Table (Descending) until correct entry is found. Then branch to tax calculation routine.
072	2	02	
073	+	-55	
074	STOI	35 46	
075	*LBL7	21 07	
076	RCLE	36 15	
077	RCLi	36 45	
078	INT	16 34	
079	X≤Y?	16-35	
080	GTO8	22 08	
081	DSZI	16 25 46	
082	GTO7	22 07	
083	*LBL8	21 08	
084	RCLi	36 45	Tax Calculation Routine
085	FRC	16 44	
086	EEX	-23	
087	5	05	
088	×	-35	
089	ENT↑	-21	
090	INT	16 34	
091	EEX	-23	
092	1	01	
093	×	-35	
094	X≠Y	-41	
095	FRC	16 44	
096	RCLE	36 15	
097	RCLi	36 45	
098	INT	16 34	
099	-	-45	
100	EEX	-23	
101	3	03	
102	×	-35	
103	×	-35	
104	+	-55	
105	RTN	24	
106	*LBLE	21 15	Mainline Calculate Alternative Tax Compute Tax on OTI Store in R_0.
107	RCLB	36 12	
108	GSB0	23 00	
109	STO0	35 00	
110	5	05	
111	EEX	-23	
112	4	04	

REGISTERS

0 Accum. Tax Amt	1 Tax Table ----------	2 ----------	3 ----------	4 ----------	5 ----------	6 ----------	7 ----------	8 ---------- 9 --------->
S0 Tax Table ----------	S1 ----------	S2 ---------->	S3 Constant 38/26/52000	S4 Ave Tax	S5	S6	S7	S8 S9 Work

A	B	C	D	E	I
Alt. Tax	OTI	C G	4 YR TI	Work	Loop Control(Int) Table Split(Frac)

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
113	RCLC	36 13	
114	X=0?	16-43	If CG = 0
115	SF2	16 21 02	Set Flag 2
116	X≤Y?	16-35	If CG < 50000
117	GTOc	22 16 13	Branch to LBL c
118	X⇄Y	-41	If C G 50000
119	4	04	Add 12500 To R_0
120	÷	-24	
121	ST+0	35-55 00	
122	2	02	Compute Tax on
123	×	-35	(OTI + 25000)
124	RCLB	36 12	Subtract from R_0
125	+	-55	
126	GSB0	23 00	
127	ST-0	35-45 00	
128	RCLB	36 12	
129	RCLC	36 13	Compute Tax on
130	2	02	$(OTI + \frac{CG}{2})$
131	÷	-24	
132	+	-55	Branch to LBL d
133	GSB0	23 00	
134	GTOd	22 16 14	
135	*LBLc	21 16 13	If CG 50000
136	4	04	Compute $\frac{CG}{4}$
137	÷	-24	
138	*LBLd	21 16 14	Add (Tax on $OTI+\frac{CG}{2}$)
139	ST+0	35-55 00	or $(\frac{CG}{4})$ to R_0
140	RCL0	36 00	
141	F2?	16 23 02	If CG=0,Alt. Tax=0
142	0	00	
143	STOA	35 11	Store Alt.Tax in R_A
144	RCLB	36 12	Calculate income
145	RCLC	36 13	averaged tax
146	2	02	Calculate $OTI+\frac{CG}{2}$
147	÷	-24	
148	+	-55	
149	RCLD	36 14	
150	X=0?	16-43	If 4 Yr TI=0
151	GTOb	22 16 12	Branch to LBL b
152	.	-62	
153	3	03	Compute .3(4Yr TI)
154	×	-35	
155	STOD	35 14	Store in R_A
156	-	-45	
157	3	03	If $(OTI+\frac{CG}{2})$ -
158	EEX	-23	.3(4YR TI)
159	3	03	< 3000
160	X⇄Y	-41	Branch to LBL b
161	X≤Y?	16-35	
162	GTOb	22 16 12	
163	5	05	Compute Tax on
164	÷	-24	$[(OTI+\frac{CG}{2} - .3(4YRTI)$
165	RCLD	36 14	$\frac{}{5}$
166	+	-55	
167	GSB0	23 00	+ 4 YR TI]
168	STO0	35 00	Store in R_0

STEP	KEY ENTRY	KEY CODE	COMMENTS
169	RCLD	36 14	Compute tax on 4YR TI
170	GSB0	23 00	Leave in R_x
171	RCL0	36 00	
172	X⇄Y	-41	Compute 4 (R_0-R_x)
173	-	-45	Add to R_0
174	4	04	
175	×	-35	
176	ST+0	35-55 00	
177	GTOa	22 16 11	
178	*LBLb	21 16 12	Leave 0 in R_0
179	0	00	
180	STO0	35 00	
181	*LBLa	21 16 11	Store
182	RCL0	36 00	Averaged Tax
183	P⇄S	16-51	in R_{54}
184	STO4	35 04	
185	P⇄S	16-51	
186	RCLD	36 14	Restore 4 YR TI
187	.	-62	in R_0
188	3	03	
189	÷	-24	
190	STOD	35 14	
191	RCLB	36 12	Calculate
192	RCLC	36 13	Regular Tax
193	2	02	Compute Tax on
194	÷	-24	$(OTI + \frac{CG}{2})$
195	+	-55	Leave in R_x
196	GSB0	23 00	Display
197	3	03	'3.00'
198	X⇄Y	-41	'Regular Tax'
199	RCLA	36 11	'Alternative Tax'
200	P⇄S	16-51	'Averaged Tax'
201	RCL4	36 04	
202	P⇄S	16-51	
203	PRST	16-14	
204	RTN	24	
205	R/S	51	

220

LABELS

A 038	B 027	C	D	E 102	0
a 177	b 174	c 131	d 134	e	1
0 001	1 025	2 045	3 040	4 051	2 Used
5 064	6 011	7 071	8 079	9 058	3

FLAGS

	ON	OFF
0	☐	☒
1	☐	☒
2	☐	☒
3	☐	☒

SET STATUS

TRIG	DISP
DEG ☒	FIX ☒
GRAD ☐	SCI ☐
RAD ☐	ENG ☐
	n 2

Program Description I

Program Title True Cost of Insurance Policy

Contributor's Name Hewlett-Packard Company

Address 1000 N. E. Circle Boulevard

City Corvallis **State** Oregon **Zip Code** 97330

Program Description, Equations, Variables The price per $1,000 of insurance protection for a given policy year may be calculated by the following formula:

$$YPT_t = \frac{(P_t+V_t-1)\ (1+i) - V_t - D_t}{(F_t-V_t)\ (.001)} \qquad (1)$$

The rate of return on the savings element in a given policy year may be calculated by the following formula:

$$i = \frac{V_t + D_t + (YPT_t)\ (F_t-V_t)\ (.001)}{P_t + V_{t-1}} - 1 \qquad (2)$$

where

YPT_t = price per $1,000 of protection in policy year t

P_t = annual premium for policy year t

V_t = cash value for policy year t

D_t = dividend for policy year t

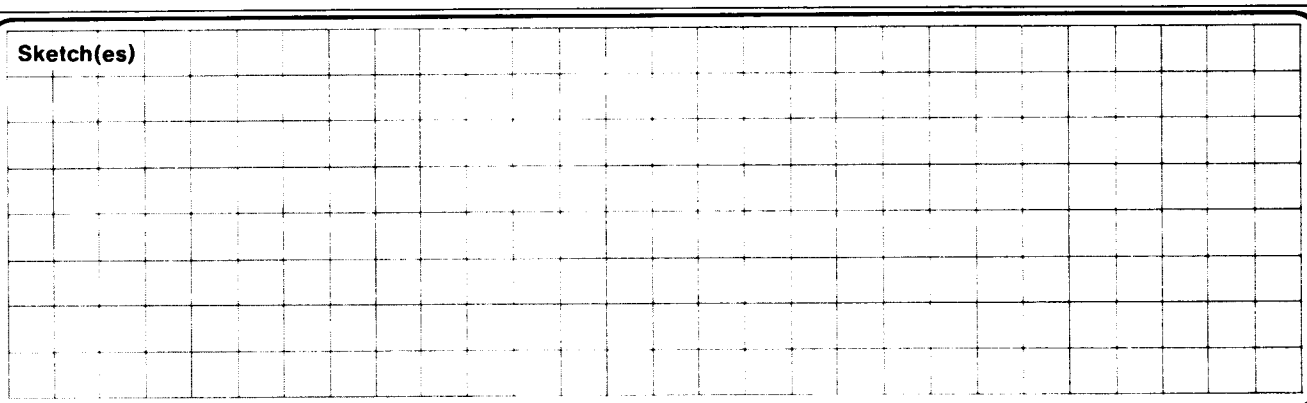
F_t = face amount for policy year t

i = rate of return on savings element, expressed as a decimal (after-tax)
~~[note that in using the program i is expressed in percent]~~

Operating Limits and Warnings You must assume a value for YPT_t (e.g., a low-cost term policy of the one-year renewable type) to calculate i; visa versa, you must assume a value for i (e.g., the interest rate you could earn on a one-year savings certificate, after tax) to calculate YPT_t.

Program Description II

Sketch(es)



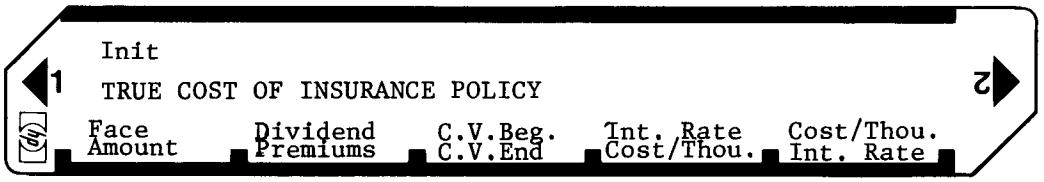
Sample Problem(s) Consider an average $50,000 face amount cash-value policy. The annual premium of $1,010 is due at the beginning of the year, and a dividend of $165 is received at the end of the policy year. Cash value of $3,302 at the beginning of the year grows to $4,104. Using formula (1) and assuming a 6% savings rate available elsewhere, the cost per $1,000 of insurance protection could be purchased for $3.00 per $1,000, the rate of return on your savings is 2.20%.

Note: even complex policies like minimum-deposit plans can be analyzed with this program. Use policy surrender values for cash-values, and the actual (after-tax) amounts for payments (premiums) and dividends.

Solution(s) Input: [f] [A]
50000 [A]
165 [Enter] 1010 [B]
3302 [Enter] 4104 [C]
For true cost per $1,000, input 6 [D] = 6.57
For savings rate, input 3 [E] = 2.20

Reference(s) Joseph M. Belth, Life Insurance - a consumer's handbook, Indiana University Press, 1973, p. 234.
This program is a translation of the HP-65 Users' Library Program #00332A submitted by Nicholas Kaiser.

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Enter Program			
2	Initialize (For each new policy analyzed)		f A	0.00
3	Input Face Amount (Total death benefit) of policy	AMT	A	AMT
4	Input Dividend to be Received at end of Current Policy Year (if any)	DIV	↑	DIV
	Input annual premium paid at beginning of current policy year	PREMIUM	B	DIV
5	Input cash (surrender) value at beginning of current policy year (if any)	C.V.BEG	↑	C.V.BEG
	Input cash value at end of current policy year	C.V.END	C	C.V.BEG
6	To see the true cost per thousand of the insurance purchased (i.e., ignoring the savings element - if any), input an assumed savings rate of interest which you know is safely available (in percent)	INT. RATE	D	Cost/Thou
7	To see the rate of interest being paid on the savings element in the policy, Input the cost per thousand of insurance which you purchase (A term policy)	COST/THOU	E	INT. RATE
8	To rerun with a new policy go to Step 2			
	To rerun with changed premium, dividends, cash value figures, just reenter A,B,or C			

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBL∝	21 16 11	
002	CLRG	16-53	Initialize
003	0	00	
004	RTN	24	
005	*LBLA	21 11	
006	STO5	35 05	Policy face amount
007	RTN	24	Store in R5
008	*LBLB	21 12	
009	STO1	35 01	Premium/Dividend
010	R↓	-31	Store annual
011	STO4	35 04	Premium in R1
012	RTN	24	
013	*LBLC	21 13	Store dividend R4
014	STO3	35 03	Cash Value
015	R↓	-31	Store cash value
016	STO2	35 02	(year end)
017	RTN	24	Store cash value
018	*LBLD	21 14	(year beginning)
019	ENT↑	-21	Calculates actual
020	1	01	cost per thousand
021	0	00	of insurance
022	0	00	assuming the rate
023	÷	-24	of return on
024	1	01	savings entered
025	+	-55	in X-reg
026	RCL1	36 01	
027	RCL2	36 02	
028	+	-55	
029	×	-35	
030	RCL3	36 03	
031	-	-45	
032	RCL4	36 04	
033	-	-45	
034	RCL5	36 05	
035	RCL3	36 03	
036	-	-45	
037	.	-62	
038	0	00	
039	0	00	
040	1	01	
041	×	-35	
042	÷	-24	
043	RTN	24	
044	*LBLE	21 15	Calculates actual
045	ENT↑	-21	rate of return on
046	RCL5	36 05	savings assuming
047	RCL3	36 03	the cost per
048	-	-45	thousand of
049	×	-35	insurance entered
050	.	-62	in X-reg
051	0	00	
052	0	00	
053	1	01	
054	×	-35	
055	RCL4	36 04	
056	+	-55	
057	RCL3	36 03	
058	+	-55	
059	RCL1	36 01	
060	RCL2	36 02	
061	+	-55	
062	÷	-24	
063	1	01	
064	-	-45	
065	1	01	
066	0	00	
067	0	00	
068	×	-35	
069	RTN	24	
070	R/S	51	

080 090 100 110

SET STATUS

FLAGS			TRIG	DISP
	ON	OFF		
0	☐	☒	DEG ☒	FIX ☒
1	☐	☒	GRAD ☐	SCI ☐
2	☐	☒	RAD ☐	ENG ☐
3	☐	☒		n 2

REGISTERS

0	1 Annual Premium	2 Cash Val Beg. Yr.	3 Cash Val Yr. End	4 Annual Dividend	5 Policy Face Amt	6	7	8	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

Program Description I

Program Title AUTOMOBILE COST / TIRE COST COMPARISON

Contributor's Name Hewlett-Packard

Address 1000 N. E. Circle Boulevard

City Corvallis **State** Oregon **Zip Code** 97330

Program Description, Equations, Variables Given initial price (P_I) and fuel economy (H = MPG, Highway; C = MPG, City) and use data:

Annual Mileage
Percent Highway Miles $\triangleq$ K = $\frac{\text{Highway Miles}}{\text{Total Miles}} \times 100$

Other inputs are length of ownership (years and tenths of years) and cost data:

Annual Interest Rate i,
Cost of Fuel $, ¢ per gallon

Equation: 1) Sinking fund $FV = PMT \left[\frac{(1 + i)^n - 1}{i} \right]$

2) Depreciation $P(N) = P(0)\ [.75]^n$

where
n = # periods i = Annual interest rate
FV = Final value PMT = Payment/Period
P(0)= Initial price P(N) = Price at N years

The program accepts price, excise tax, and estimated life (mileage) for up to 3 different tires and computes the best buy based on the number of miles expected for dollar spent. Best buy is indicated by displaying entered tire price for identification and miles per dollar as a single split display. Single tire price, miles per dollar, and four tire set price can also be recalled for each case.

Equations solved are:

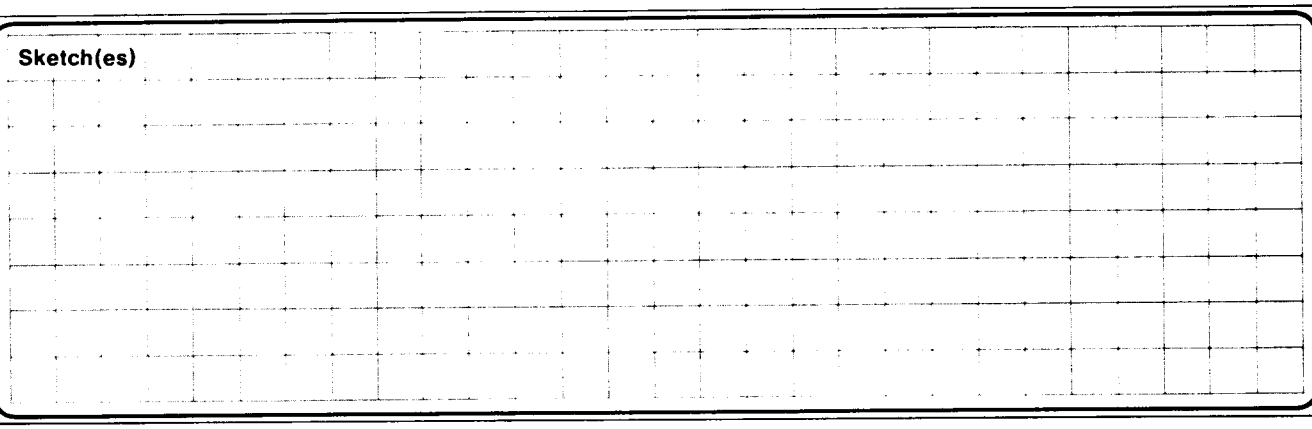
1) 4 X (Excise Tax & Tire Price) = 4 Tire Set Price
2) $\frac{\text{Estimated Life in Miles}}{\text{4 Tire Set Price}}$ = Miles/Dollar
3) Best Buy = Greatest Miles/Dollar Ratio

OPERATING LIMITS AND WARNINGS

Percentages for highway mileage and interest rate are entered as whole numbers between 0 and 100.

Program Description II

Sketch(es)



Sample Problem(s) 1. Compute the net cost difference at the end of 3 years of ownership of two candidate autos:

1) The "Wombat I" with P_1 = $4,000
HWY= 25 city = 15

2) The "Lightfoot 2" with P_2 = $3,000
HWY = 30 city = 20

Assume use data as follow - annual mileage 12,000 miles/year, and 50% use is HWY type. Cost data is taken as : Interest rate 6%
Fuel at $0.60/ gal

2. Find the best buy of the 3 tires below:

#1 Price $54.22; Excise Tax $2.52; Estimated Life 36,000 Miles
#2 Price $36.57; Excise Tax $1.96; Estimated Life 25,000 Miles
#3 Price $29.88; Excise Tax $1.22; Estimated Life 17,000 Miles

SOLUTION(S):

1.	4000 [↑] 25 [↑] 15 [A] →	4000
	3000 [↑] 30 [↑] 20 [B] →	3000
	12000 [↑] 50 [C] →	10
	3 [D] →	421.88
	6 [↑] 0.60 [E] →	1010.82
2.	Enter 2.52 [ENT↑]	2.52
	" 54.22 [ENT↑]	54.22
	" 36000 [f] [A]	226.96 (4 Tire Cost)
	" 1.96 [ENT↑]	1.96
	36.57 [ENT↑]	36.57
	25000 [f] [B]	154.12 (4 Tire Cost)
	1.22 [ENT↑]	1.22
	29.88 [ENT↑]	29.88
	17000 [f] [C]	124.40 (4 Tire Cost)

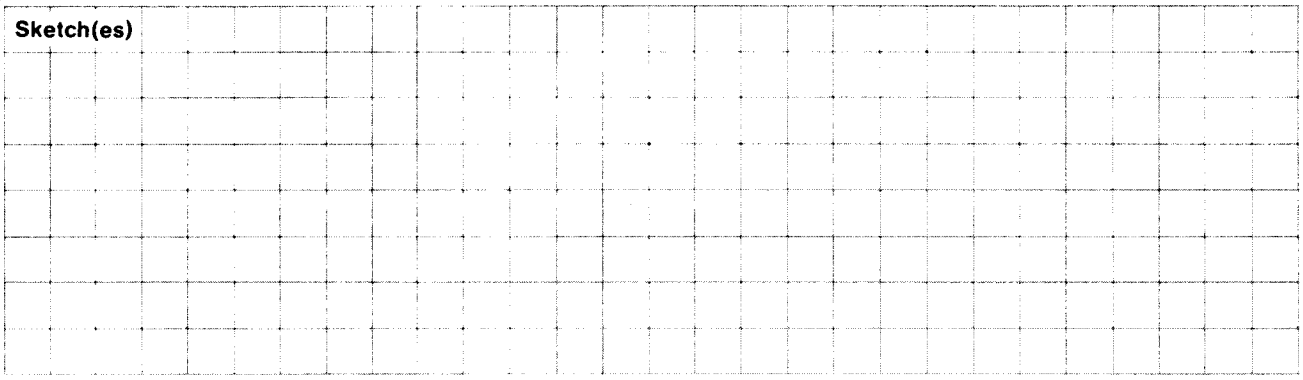
(Continued on following page)

Reference(s) This program is a modification of the Users' Library Program #04638A submitted by Robert S. Siebert.

This program is a modification of the Users' Library Program #03428A submitted by Robert A. Plack.

Program Description 11

Sketch(es)



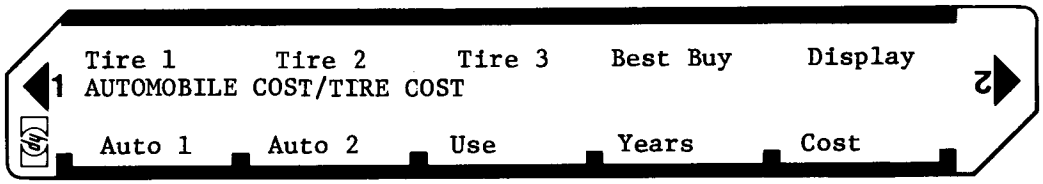
Sample Problem(s)

Solution(s)

```
  Compute and Display Best Buy     [f] [D]      36.5700162  [Tire 2 best buy] 162 Mi./$
                      Display      [f] [E]      54.2200159
                                               226.96        Tire 1
                                                36.5700162
                                               154.12        Tire 2
                                                29.8800137
                                               124.40        Tire 3
```

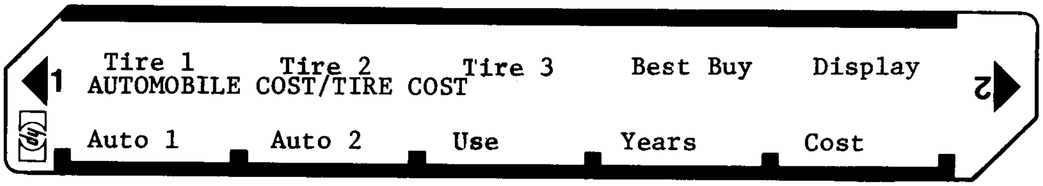
Reference(s)

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS		OUTPUT DATA/UNITS
1	Load side 1 and side 2				
2	For automobile cost comparison, go to step 3				
	For tire cost comparison, go to step 9				
3	For automobile #1:				
	Enter price:	$P1	ENT↑		
	highway mpg:	MPG	ENT↑		
	city mpg:	MPG	A		P_1
4	For automobile # 2:				
	Enter price:	$P2	ENT↑		
	highway mpg:	MPG	ENT↑		
	city mpg:	MPG	B		P_2
5	Enter use data:				
	annual mileage	MILES	ENT↑		
	percent highway	%	C		Δ Gal/Mo.
	(0 < K ⩽ 100)				
6	Years of service	Years	D		Future Val.
7	Enter costs:				of P_1-P_2
	Interest rate	i	ENT↑		
	Fuel Cost	$/Gal.	E		Net Cost
8	To re-run, change desired step and reenter				
	all higher numbered steps.				
9	For tire #1:				
	Enter tire excise tax	$	ENT↑		
	single tire price	$	ENT↑		
	tire estimated life	MILES	f	A	4 Tire Pr.
10	For tire #2:				
	Enter tire excise tax	$	ENT↑		
	single tire price	$	ENT↑		
	tire estimated life	MILES	f	B	4 Tire Pr.
11	(Optional: For tire #3)				
	Enter tire excise tax	$	ENT↑		
	single tire price	$	ENT↑		
	tire estimated life	MILES	f	C	4 Tire Pr.
12	Compute best buy based on miles/dollar.		f	D	$-Miles/$
	(Read split display as follows:				
	- - . - - - - - -				
	single tire price / miles/$				
	(Continued following page)				

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
13	To display (Print on HP-97)		f E	Price/Mi./$
				4 Tire $
				Pr. + Mi/$
				4 Tire $
	(Only if data entered for 3rd tire)			Pr.+Mi./$
				4 Tire $
14	To begin new calculation, go to step 9			
			f E	

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBLA	21 11	
002	STO1	35 01	
003	-	-45	
004	STO2	35 02	
005	R↓	-31	
006	RTN	24	Price 1
007	*LBLB	21 12	
008	STO3	35 03	
009	-	-45	
010	STO4	35 04	
011	R↓	-31	
012	-	-45	
013	STO5	35 05	
014	-	-45	
015	RTN	24	Price 2
016	*LBLC	21 13	
017	1	01	
018	0	00	
019	0	00	
020	÷	-24	
021	STx4	35-35 04	
022	RCL2	36 02	
023	x	-35	
024	RCL1	36 01	
025	+	-55	
026	1/X	52	
027	RCL4	36 04	
028	RCL3	36 03	
029	+	-55	
030	1/X	52	
031	-	-45	
032	x	-35	
033	1	01	
034	2	02	
035	÷	-24	
036	STO6	35 06	Gallons/Mo.Diff
037	RTN	24	
038	*LBLD	21 14	
039	STO9	35 09	
040	1	01	
041	2	02	
042	x	-35	
043	STO7	35 07	
044	.	-62	
045	7	07	
046	5	05	
047	RCL9	36 09	
048	Y^x	31	
049	RCL5	36 05	
050	x	-35	
051	STO9	35 09	Remaining Value of Cost Difference
052	RTN	24	
053	*LBLE	21 15	
054	RCL6	36 06	
055	x	-35	
056	X⇄Y	-41	
057	1	01	
058	2	02	
059	0	00	
060	0	00	
061	÷	-24	
062	STO8	35 08	
063	1	01	
064	+	-55	
065	RCL7	36 07	
066	Y^x	31	
067	STx5	35-35 05	
068	STO2	35 02	
069	1	01	
070	-	-45	
071	RCL8	36 08	
072	÷	-24	
073	x	-35	
074	RCL5	36 05	
075	+	-55	
076	RCL9	36 09	
077	-	-45	
078	RCL5	36 05	Restores 5.
079	RCL2	36 02	
080	÷	-24	
081	STO5	35 05	
082	X⇄Y	-41	Net Cost Difference
083	RTN	24	
084	*LBLa	21 16 11	Initialize
085	DSP2	-63 02	
086	CF1	16 22 01	
087	SF2	16 21 02	
088	CLRG	16-53	
089	STO7	35 07	
090	R↓	-31	#1 Tire Life
091	STO1	35 01	#1 Tire Price
092	STO4	35 04	Fill to Prevent
093	STO6	35 06	Division by Zero
094	+	-55	if only 2 cases en.
095	4	04	Compute #1 Tire
096	x	-35	4 Tire Price
097	STO2	35 02	
098	RTN	24	
099	*LBLb	21 16 12	
100	STO8	35 08	#2 Tire Life
101	R↓	-31	
102	STO3	35 03	#2 Tire Price
103	+	-55	Compute #2 Tire
104	4	04	4 Tire Price
105	x	-35	
106	STO4	35 04	
107	RTN	24	
108	*LBLc	21 16 13	
109	CF2	16 22 02	
110	STO9	35 09	#3 Tire Life
111	R↓	-31	
112	STO5	35 05	#3 Tire Price

REGISTERS

0	1 Used Tire C1	2 Used Tire T1	3 Used TC2	4 Used TT2	5 $P_1 - P_2$ TC3	6 ΔMPG TT3	7 # Mo. TL1	8 L. TL2	9 Final Pr TL3
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
113	+	-55	
114	4	04	
115	x	-35	
116	STO6	35 06	
117	RTN	24	
118	*LBLd	21 16 14	Compute #3 Tire 4 Tire Price
119	RCL7	36 07	
120	RCL2	36 02	
121	GSB0	23 00	
122	STO7	35 07	
123	ST+1	35-55 01	
124	RCL8	36 08	
125	RCL4	36 04	
126	GSB0	23 00	
127	STO8	35 08	
128	ST+3	35-55 03	
129	RCL9	36 09	
130	RCL6	36 06	
131	GSB0	23 00	
132	STO9	35 09	
133	ST+5	35-55 05	
134	DSP7	-63 07	
135	RCL9	36 09	Compute Miles Per Dollar
136	RCL8	36 08	
137	X>Y?	16-34	
138	GTO1	22 01	
139	X≷Y	-41	
140	SF1	16 21 01	
141	*LBL1	21 01	
142	RCL7	36 07	
143	X>Y?	16-34	
144	GTO2	22 02	
145	F1?	16 23 01	
146	GTO3	22 03	
147	RCL3	36 03	
148	RTN	24	Display Best Buy
149	*LBL2	21 02	
150	RCL1	36 01	
151	RTN	24	
152	*LBL3	21 03	
153	RCL5	36 05	
154	RTN	24	
155	*LBLe	21 16 15	Display Mode
156	DSP7	-63 07	
157	RCL1	36 01	#1 Price - Miles/$
158	PRTX	-14	
159	DSP2	-63 02	
160	RCL2	36 02	#1 4 Tire Price
161	PRTX	-14	
162	SPC	16-11	
163	RCL3	36 03	#2 Price - Miles/$
164	DSP7	-63 07	
165	PRTX	-14	
166	RCL4	36 04	#2 4 Tire Price
167	DSP2	-63 02	
168	PRTX	-14	
169	SPC	16-11	
170	F2?	16 23 02	
171	RTN	24	
172	DSP7	-63 07	
173	RCL5	36 05	#3 Price - Miles/$
174	PRTX	-14	
175	DSP2	-63 02	
176	RCL6	36 06	#3 4 Tire Price
177	PRTX	-14	
178	RTN	24	
179	*LBL0	21 00	Miles/Dollar Sub Routine
180	÷	-24	
181	EEX	-23	
182	7	07	
183	÷	-24	
184	RTN	24	
185	R/S	51	

190 200 210 220

LABELS

A Auto 1	B Auto 2	C Use	D # Years	E Costs	0
a Tire 1	b Tire 2	c Tire 3	d Best	e Display	1 Used
0 Used	1 Used	2 Used	3 Used	4	2 Used
5	6	7	8	9	3

FLAGS

SET STATUS

FLAGS	ON	OFF
0	☐	☒
1	☐	☒
2	☐	☒
3	☐	☒

TRIG		DISP	
DEG	☐	FIX	☐
GRAD	☐	SCI	☐
RAD	☐	ENG	☐
		n	____

Program Description I

Program Title COMPARISON SHOPPING

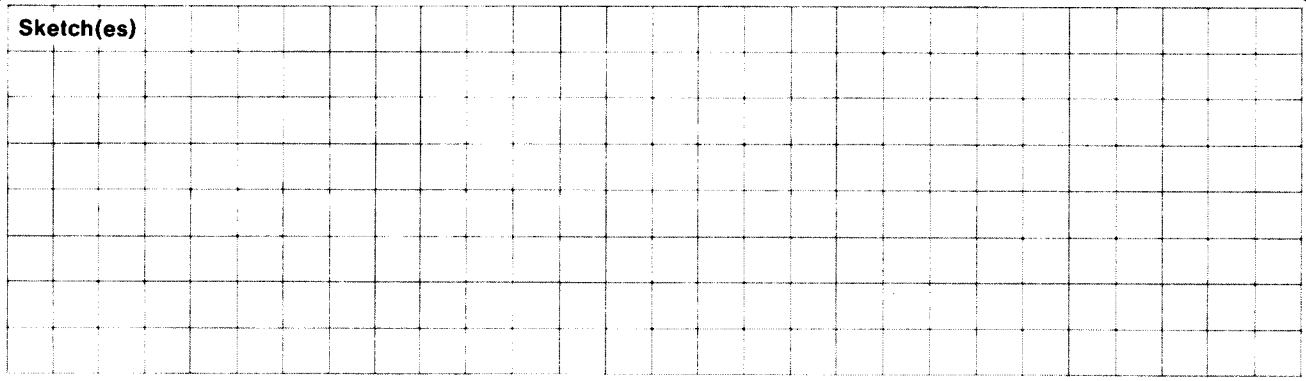
Contributor's Name Hewlett-Packard Company
Address 1000 N. E. Circle Boulevard
City Corvallis **State** Oregon **Zip Code** 97330

Program Description, Equations, Variables Competing manufacturers often package their products in various weights or volumes, so that in stores that lack some form of "unit pricing", consumers have a difficult time deciding which brand is actually cheapest. This program overcomes that difficulty. The price and amount of each item is entered successively. The program converts to cost/amount, which is compared to that of the item which was cheapest up to that point. In case erroneous input is entered, the second least expensive item is retained along with the least. This can be recalled if the output data cannot be associated with any product. Output data identifies the least expensive product by its cost, amount, and gives the cost/amount. A counter lists the number of items input so far with each new entry.

Operating Limits and Warnings None

Program Description II

Sketch(es)



Sample Problem(s) Find the lowest price amoung the following products:

Item	Price	Weight
Ketchup 1	$ .85	16.5 oz.
Ketchup 2	.69	14.75 oz.
Catsup	.78	16.0 oz.

Solution(s)

[A]
.85 [Enter↑] 16.5 [B]
.69 [Enter↑] 14.75 [B]
.78 [Enter↑] 16.0 [B]
[C] 14.75 weight
[R/S] 0.69 price
[R/S] 0.468 price/weight

Reference(s) This program is a translation of the HP-65 Users' Library Program # 01258A submitted by Randal Lee O'Toole.

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Enter Program			
2	Initialize for new data		A	
3	Enter price of first item	Price	Ent↑	
4	Enter amount of first item	Amount	B	# Item Ent.
5	For second and any further items, go to 3.			
6	To see amount of cheapest item		C	Amount
	To see price of cheapest item (optional)		R/S	Price
	To see price/amount of cheapest item (option)		R/S	Pr./Amt.
7	To begin a new series of items, go to 2			
8	If erroneous data is entered and now claims		D	
	to be the cheapest item, D will recall the			
	item previously considered cheapest.			

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBL B	21 12	Entry of data
002	F1?	16 23 01	First entry is
003	GTO 0	22 00	treated seperately
004	STO 1	35 01	Amount
005	STO 5	35 05	
006	÷	-24	
007	STO 2	35 02	Cost/Amount
008	STO 4	35 04	
009	SF 1	16 21 01	Flag to show further
010	GTO 1	22 01	ent. aren't first
011	*LBL 0	21 00	Second and further
012	STO 3	35 03	entries treated
013	÷	-24	here
014	STO 6	35 06	
015	RCL 2	36 02	
016	X≤Y?	16-35	Comparison cost/amt
017	GTO 1	22 01	
018	STO 4	35 04	
019	RCL 1	36 01	
020	STO 5	35 05	
021	RCL 6	36 06	
022	STO 2	35 02	
023	RCL 3	36 03	
024	STO 1	35 01	
025	*LBL 1	21 01	
026	DSZI	16 25 46	Counter of entries
027	RCL I	36 46	
028	CHS	-22	
029	DSP 0	-63 00	
030	RTN	24	
031	*LBL C	21 13	Identification of
032	DSP 2	-63 02	least expen. item
033	RCL 1	36 01	Amount
034	RTN	24	
035	RCL 1	36 01	Cost
036	RCL 2	36 02	
037	×	-35	
038	R/S	51	
039	RCL 2	36 02	Cost/Amount
040	DSP 4	-63 04	
041	R/S	51	
042	*LBL A	21 11	Signal new products
043	CF 1	16 22 01	being compared
044	CLRG	16-53	
045	CLX	-51	
046	RTN	24	
047	*LBL D	21 14	Recalls second
048	RCL 4	36 04	least expensive
049	STO 2	35 02	data
050	RCL 5	36 05	
051	STO 1	35 01	
052	RTN	24	
053	R/S	51	

STEP	KEY ENTRY	KEY CODE	COMMENTS
060			
070			
080			
090			
100			
110			

SET STATUS			
FLAGS		TRIG	DISP
	ON OFF		
0	☐ ☒	DEG ☒	FIX ☒
1	☐ ☒	GRAD ☐	SCI ☐
2	☐ ☒	RAD ☐	ENG ☐
3	☐ ☐		n 2

REGISTERS

0	1 Cheapest Amount	2 Cheapest Cost/Amt	3 Competing Amount	4 Second Ch CT/Amt	5 Second cheap amt	6 Competing Cost/Amt.	7	8	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

Program Description I

Program Title TIME & CHARGES RUNNING TOTAL

Contributor's Name THOMAS M. COX JR.
Address 7544 MAPLE LEAF LANE
City SACRAMENTO State CALIF Zip Code 95828

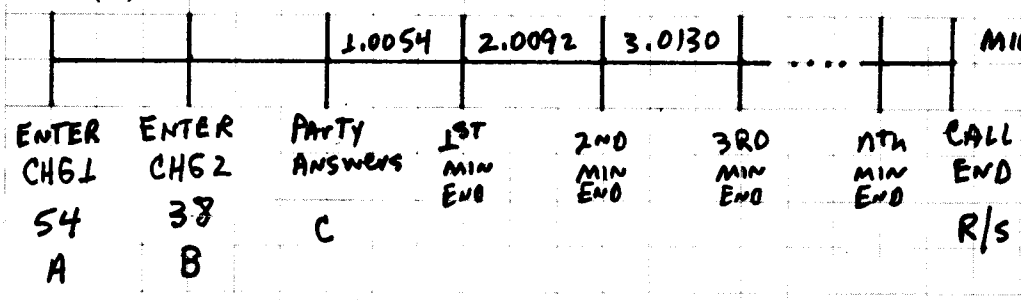
Program Description, Equations, Variables PROGRAM USES A TEN SECOND LOOP TO TIME A TELEPHONE CALL. THE LOOP PAUSES FIRST TO DISPLAY TENS OF SECONDS, then FIVE PAUSES DISPLAY MINUTE ONE IS INTO LEFT of DECIMAL POINT AND CHARGE IN CENTS TO THE RIGHT, then 30 "+" INSTRUCTIONS ARE USED FOR CALIBRATION PURPOSES (figure 50 MS. PER ADD PERFORMED).

PROGRAM need NOT be MORE ACCURATE than THE MINUTE PLUS FIFTY SECONDS — IT needs ONLY TO TELL WHAT MINUTE ONE IS INTO FOR BILLING PURPOSES. A TIMING GAIN of ONE SECOND IN FIVE MINUTES IS EASILY ACHIEVED AND DEEMED MORE THAN ADEQUATE FOR LONG DISTANCE CALLS LASTING less than TWO HOURS. (FIVE HOURS before DISPLAY WOULD BE WRONG IF CALLING ON A WEEKEND WITHIN 48 STATES.) PROGRAM EASILY MODIFIED TO HANDLE CALLS COSTING OVER $100 IF DESIRED (ADD 0 step 72, DSPS STEP 5).

Operating Limits and Warnings (see Above)

Program Description II

Sketch(es)



Sample Problem(s) CHARGES DURING NORMAL BUSINESS HOURS FOR A PHONE CALL FROM SACRAMENTO TO NEW YORK CITY IS 54¢ for first MINUTE AND 38¢ FOR EACH SUCCEEDING MINUTE. CAUSE THE CALCULATOR TO DISPLAY TIME AND CHARGES AFTER PARTY ANSWERS, UPDATED AT ONE MINUTE INTERVALS.

Solution(s) Key strokes:

54 A → 1.0054

38 B → 1.0038 PAUSE, 1.0054 HALT.

PARTY ANSWERS C → 1.0054, 2.0092, 3.0130, etc.

Reference(s)

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	LOAD SIDE 1			
2	INPUT CHARGE 1 (2 DIGIT INTEGER)	CHG 1	A	MIN. CHG 1
3	INPUT CHARGE 2 "	CHG 2	B	MIN. CHG 1
4	WHEN PARTY CALLED ANSWERS, RUN.		C	MIN. CENTS
5	WHEN CALL ENDS, STOP.		R/S	MIN. TOTAL
	CALIBRATION ROUTINE: TIME PROGRAM A FEW MINUTES; IF FAST INSERT "+" INSTRUCTION AT STEP 31; IF SLOW DELETE "+". EACH "+" WILL CHANGE TIMING APPROX. .3 SEC PER MINUTE (6 X .05 = .3).			
	EXAMPLE: 1.0054 MINUTE, TEN SECOND LOOP NUMBER 1 (STEP 19) TIME 3:37:20 TO 15.0586 LOOP NUMBER 5 TIME 3:53:16, CALCULATOR GAINED 44 SECONDS IN 56 SEC, GAINING 1/20 sec per sec sec per MIN, INSERT 10 "+" INSTRUCTIONS (3/.3) TO OBTAIN A GAIN of less THAN 1 SECOND IN FIVE MINUTES.	13 MIN OR 3		

67 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	f LBL A	31 25 11	STORE CHG 1
	STO A	33 11	for Reference
	f GSB 1	31 22 01	
	STO 1	33 01	1ST MIN. CHG
	DSP 4	23 04	
	h RTN	35 22	
	f LBL B	31 25 12	STORE CHG 2
	STO B	33 12	for Reference
	f GSB 1	31 22 01	
010	STO 2	33 02	
	h PAUSE	35 72	2ND MIN. CHG
	RCL 1	34 01	1ST MIN. CHG
	h RTN	35 22	
	f LBL C	31 25 13	RUN
	0	00	
	h STI	35 33	
	f LBL 2	31 25 02	
	h RCI	35 34	
	h PAUSE	35 72	TENS of SECONDS
020	RCL 1	34 01	
	h PAUSE	35 72	
	h PAUSE	35 72	
	h PAUSE	35 72	
	h PAUSE	35 72	
	h PAUSE	35 72	
	1	01	
	ENTER	41	
	ENTER	41	
	ENTER	41	
030	0	00	
	+	61	
	|	|	30 EACH "+" INSTRUCTIONS
040			
050			

STEP	KEY ENTRY	KEY CODE	COMMENTS
	|	|	
060	+	61	
	STO 3	33 03	ADDITIONS COUNT
	f ISG	31 34	
	h RCI	35 34	
	6	06	
	g x≠y?	32 81	
	GTO 2	22 02	
	RCL 2	34 02	
	STO + 1	33 61 01	
	GTO C	22 13	
070	f LBL 1	31 25 01	INITIALIZE ROUTINE
	1	01	
	0	00	
	0	00	
	0	00	
	0	00	
	÷	81	
	1	01	
	+	61	
	h RTN	35 22	
080			
090			
100			
110			

FLAGS	SET STATUS		
0	FLAGS	TRIG	DISP
1	ON OFF		
2	0 ☐ ☐	DEG ☐	FIX ☐
3	1 ☐ ☐	GRAD ☐	SCI ☐
	2 ☐ ☐	RAD ☐	ENG ☐
	3 ☐ ☐		n____

REGISTERS

0	1	2	3	4	5	6	7	8	9
	TIME. CHG	MIN. CENTS	#ADDS						
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I
CHARGE 1	CHARGE 2				#LOOP

Program Description I

Program Title: Reconcile Checking Account

Contributor's Name: Geoffrey Kidd
Address: 1514 Oxford Street Apt #301
City: Berkeley, State: CA Zip Code: 94709

Program Description, Equations, Variables: This program serves as an aid in reconciling personal checkbook tallies to bank statements. Inputs are outstanding(uncancelled) checks, outstanding deposits(deposits after the statement closing date), and bank statement balance. Outputs are: final balance, which should agree with the personal checkbook tally, sum and total number of outstanding checks, and sum and number of deposits.

Formula: $FB = SB + \sum_{i=1}^{m_D} D_i - \sum_{i=1}^{m_C} C_i$

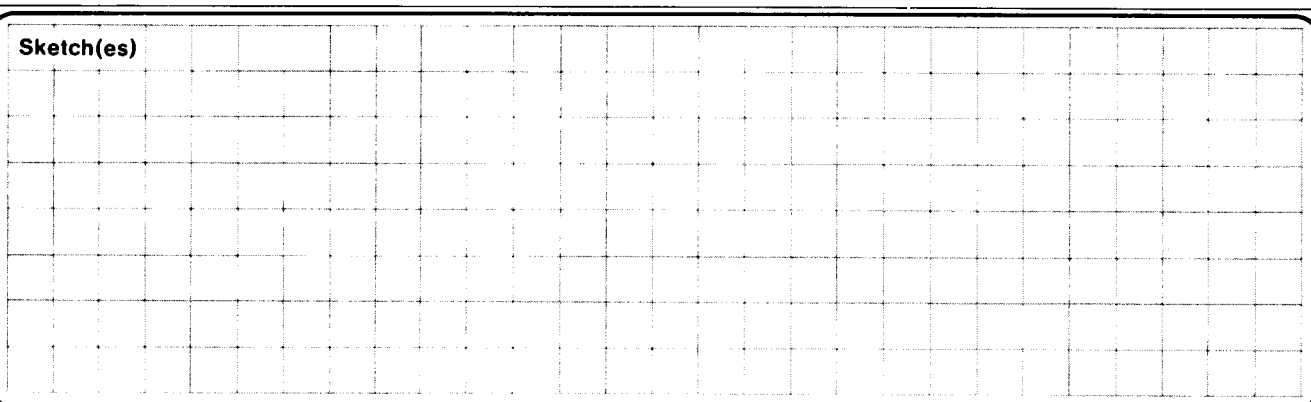
where FB=final balance, SB=bank statement balance, D_i=Outstanding deposit number i, C_i=Outstanding check number i, m_D=Number of outstanding deposits, n_C=Number of outstanding checks.

The checkbook balance,, then, is equal to the bank statement balance plus deposits made after the statement closing date minus checks not received at the bank before the closing date.

Operating Limits and Warnings: All statement service charges should be subtracted from the checkbook tally before reconciling. Also, insure that all checks which have been cancelled are accounted for in the checkbook tally and are not still listed as outstanding.

Program Description II

Sketch(es)



Sample Problem(s)

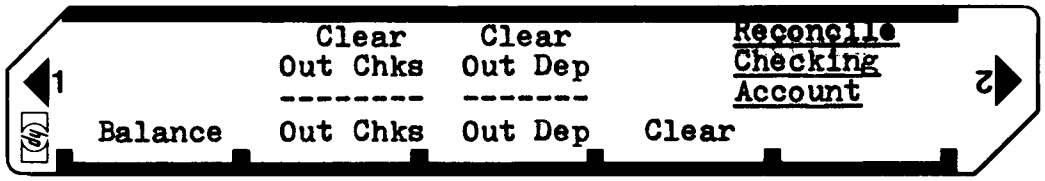
As an example, suppose the bank statement balance was $432.96. You had outstanding checks of $47.82, $5.63, $25.00, $36.47, and $96.02. The outstanding deposits are $100.00 and $256.03. Compute the current checkbook balance, total of outstanding checks, and total of outstanding deposits.

Solution(s)

```
[D] 432.96[A]
47.82[B] 5.63[B] 25[B] 36.47[B] 96.02[B]
100[C] 256.03[C]
[A] ----------------------------------------->   578.05 Current checkbook balance
[B] ----------------------------------------->   210.94 Total outstanding checks
[C] ----------------------------------------->   356.03 Total outstanding deposits
```

Reference(s)

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Enter program			
2.	Initialize		D	0.00
3.	Input Statement Balance	SB	A	SB
4.	Repeat 4. for each check.	C_1-C_n	B	C_1-C_n
5.	Repeat 5. for each deposit.	D_1-Dn	C	D_1-D_n
6.	Compute final balance		A	FB
7.	Clear outstanding checks		f b	0.00
	or outstanding deposits		f c	0.00
8.	Recall sum of outstanding checks		B	ΣC_i
	or sum of outstanding deposits		C	ΣD_i
9.	To add additional checks, go to 4.			
10.	To add additional deposits, go to 5.			
11.	For new case, go to 2.			

67 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	f LBL D	31 25 14	(Initialize)
	f CL REG	31 43	Clear reg + stack
	CLx	44	
	E↑	41	
	E↑	41	
	E↑	41	
	R RTN	35 22	
	f LBL A	31 25 11	(Balance)
	R F? 3	35 71 03	If new entry,
010	GTO 0	22 00	branch
	RCL 0	34 00	Calculate FB
	RCL 1	34 01	
	−	51	
	RCL 2	34 02	
	+	61	
	R RTN	35 22	
	f LBL 0	31 25 00	
	STO 0	33 00	STO SB
	R RTN	35 22	
020	f LBL B	31 25 12	(Sum chks)
	R F? 3	35 71 03	If new entry,
	GTO 1	22 01	branch
	RCL 3	34 03	else, Set up for display
	RCL 1	34 01	Σc, # checks
	GTO 3	22 03	
	f LBL 1	31 25 01	
	STO + 1	33 61 01	Augment Σc,
	E↑	41	# checks
	1	01	
030	STO + 3	33 61 03	
	R RV	35 53	
	R RTN	35 22	
	f LBL C	31 25 13	(Sum Dep)
	R F? 3	35 71 03	If new entry,
	GTO 2	22 02	branch
	RCL 4	34 04	else display
	RCL 2	34 02	Σ D, # deposits
	GTO 3	22 03	
	f LBL 2	31 25 02	
040	STO + 2	33 61 02	Augment Σ D,
	E↑	41	# deposits
	1	01	
	STO + 4	33 61 04	
	R RV	35 53	
	R RTN	35 22	
	g LBL b	32 25 12	(Clear out chk)
	CLx	44	Clear Σ C,
	STO 1	33 01	# checks
	STO 3	33 03	
050	R RTN	35 22	
	g LBL C	32 25 13	(Clear out dep)
	CLx	44	Clear Σ D,
	STO 2	33 02	# checks
	STO 4	33 04	
	R RTN	35 22	
	f LBL 3	31 25 03	(Routine for display)

STEP	KEY ENTRY	KEY CODE	COMMENTS
	f −x−	31 84	
	R x≷y	35 52	
	f −x²	31 84	
060	R x≷y	35 52	
	R RTN	35 22	
	R/S	84	
070			
080			
090			
100			
110			

Labels used:
A, B, C, D
b, c
0, 1, 2, 3

Set status
all flags off
DEG mode
FIX display
n = 2

REGISTERS

0	1	2	3	4	5	6	7	8	9
SB	ΣC_i	ΣD_i	# chks	# dep					
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

Program Description I

Program Title SAVINGS ACCOUNT COMPOUNDED DAILY

Contributor's Name Hewlett-Packard Users' Library
Address 1000 N.E. Circle Blvd.
City Corvallis **State** OR **Zip Code** 97330

Program Description, Equations, Variables This program uses the formula

$B_{new} = B_{old} (1 + i/36525)^N$

where B_{new} = new balance

B_{old} = old balance

i = interest (in %)

N = Number of days elapsed

N is computed using the following calendar formula:

D(m,d,y) = number of days since a certain fixed date in antiquity

$= d + [30.6\ f(m)] + [365.25\ g(y,m)]$,

where $f(m) = \begin{cases} m + 13 \text{ if } m = 1 \text{ or } 2 \\ m + 1 \text{ if } m > 2 \end{cases}$ $g(y,m) = \begin{cases} y - 1 & m = 1, 2 \\ y & m > 2 \end{cases}$ [ ] = f INT

N = new D - previous D.

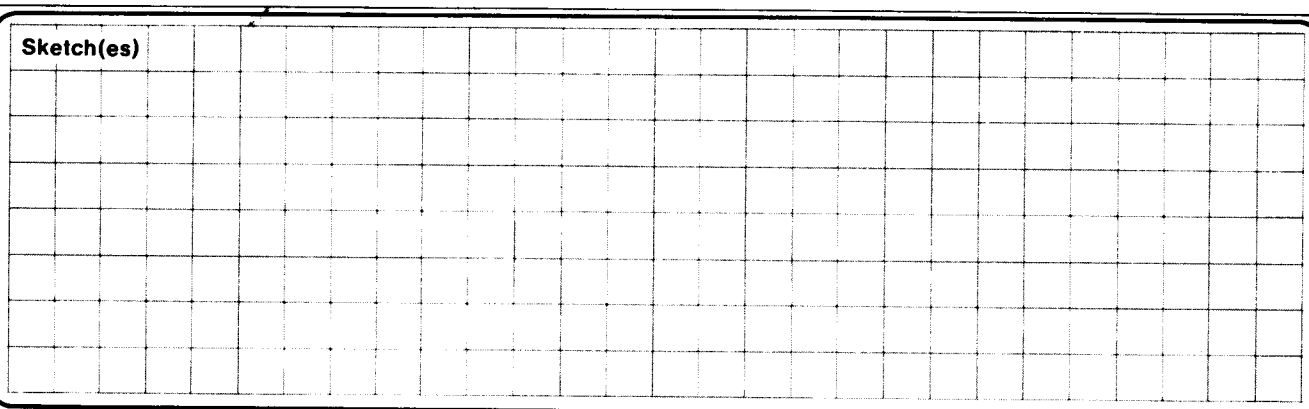
Dates are coded mm.dd.

Operating Limits and Warnings

1. The calendar routine is valid from March 1, 1900 through February 28, 2100; however, the error for dates outside of this range is minute.
2. Interest rates are based on a 365 1/4-day year, which may differ slightly from what banks use. It should even out in the long run, however.
3. Error messages are not given for illegal date entries.

Program Description 11

Sketch(es)



Sample Problem(s)

Compute interest on this account:

Starting year 1976, interest 5.25%

1. July 1 deposit $1500.00
2. September 22 deposit $154.50
3. September 27 withdraw $148.00
4. October 15 deposit $133.33
5. October 16 withdraw $15.87
6. November 5 withdraw $180.00
7. February 12 deposit $40.00
8. July 7 withdraw $300.00
9. November 22 withdraw $50.00
10. May 5 deposit $173.21
11. October 22 withdraw $400.00

Find the balance at the end of each transaction and the total interest at the end.

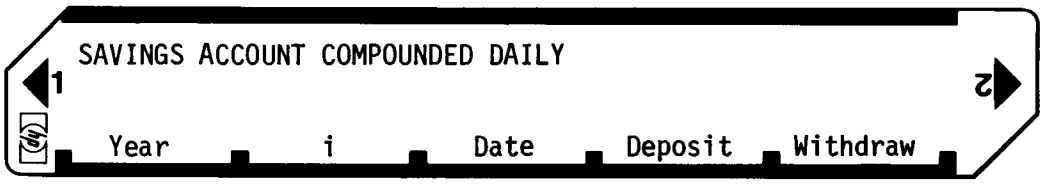
Solution(s)

1. 1500.00
2. 1672.50
3. 1525.70
4. 1662.99
5. 1647.35
6. 1472.10
7. 1533.19
8. 1265.48
9. 1240.83
10. 1443.64
11. 1079.34

Total interest = 172.17

Reference(s)

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS		OUTPUT DATA/UNITS
1	Enter program.				
2	Clear registers.		f	CL REG	
3	Enter starting year.	y	A		
4	Enter interest rate.	i/in %	B		
5	Enter starting date.	mm.dd	C		
6	Deposit starting balance.	$	D		Balance
7	Enter next transaction date.	mm.dd	C		New Balance
8	Enter all deposits and withdrawals for that				
	date.	Deposit	D		New Balance
	and/or	Withdrawal	E		New Balance
	Repeat steps 7 and 8 for each date.				
9	Enter final date.	mm.dd	C		Final Balance
	To see total interest anytime after step 5		RCL	0	Total Interest
	To manually change the year, perform step 3				
	before step 7. Note: As long as the period				
	between any two consecutive transactions is				
	less than a year, the year is automatically				
	changed. So it is necessary to manually change				
	the year only if the account is dormant for				
	over a year.				
	To change the interest rate, perform step 4				
	after step 7.				
	For a new account go back to step 2.				

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	001 *LBLA	21 11	Year → R_3
	002 STO3	35 03	
	003 0	00	
	004 STO4	35 04	Clear date
	005 RTN	24	
	006 *LBLB	21 12	
	007 3	03	
	008 6	06	
	009 5	05	
010	010 2	02	
	011 5	05	
	012 STO6	35 06	
	013 ÷	-24	
	014 1	01	
	015 +	-55	
	016 LN	32	LN (1 + i/36525)
	017 STO5	35 05	→ R_5
	018 0	00	
	019 RTN	24	
020	020 *LBLC	21 13	
	021 RCL4	36 04	
	022 X≠Y	-41	
	023 STO4	35 04	Compare date with previous date. If new date is smaller, add 1 to year.
	024 -	-45	
	025 0	00	
	026 X≤Y?	16-35	
	027 1	01	
	028 RCL3	36 03	
	029 +	-55	
030	030 STO3	35 03	
	031 3	03	
	032 RCL4	36 04	
	033 1	01	
	034 +	-55	
	035 INT	16 34	
	036 X>Y?	16-34	If month is Jan. or Feb., add 12 to m and subtract 1 from y.
	037 GTO0	22 00	
	038 9	09	
	039 +	-55	
040	040 +	-55	
	041 X≠Y	-41	
	042 1	01	
	043 -	-45	
	044 *LBL9	21 09	
	045 RCL6	36 06	
	046 ×	-35	
	047 EEX	-23	
	048 2	02	
	049 ÷	-24	
050	050 INT	16 34	[365.25y]
	051 RCL4	36 04	
	052 FRC	16 44	
	053 EEX	-23	
	054 2	02	
	055 ×	-35	
	056 +	-55	J

STEP	KEY ENTRY	KEY CODE	COMMENTS
	057 X≠Y	-41	m
	058 3	03	
	059 0	00	
060	060 .	-62	
	061 6	06	
	062 ×	-35	
	063 INT	16 34	[30.6 m]
	064 +	-55	D = d + [30.6m]
	065 RCL2	36 02	+ [365.25y]
	066 X≠Y	-41	
	067 STO2	35 02	
	068 -	-45	
	069 CHS	-22	N = difference
070	070 RCL5	36 05	LN (1 + i/36525)
	071 ×	-35	$(1 + i/36525)^N$
	072 e^x	33	
	073 1	01	
	074 -	-45	
	075 RCL1	36 01	Old balance
	076 ×	-35	New interest
	077 ST+0	35-55 00	Total interest
	078 CHS	-22	Negate following
	079 *LBLE	21 15	CHS
080	080 CHS	-22	Withdraw
	081 *LBLD	21 14	
	082 ST+1	35-55 01	Deposit
	083 RCL1	36 01	New balance
	084 RTN	24	
	085 *LBL0	21 00	
	086 R↑	16-31	
	087 GTO9	22 09	
	088 R/S	51	
090			
100			

FLAGS

0	
1	
2	
3	

SET STATUS

FLAGS	ON	OFF
0	☐	☒
1	☐	☒
2	☐	☒
3	☐	☒

TRIG	
DEG	☒
GRAD	☐
RAD	☐

DISP	
FIX	☒
SCI	☐
ENG	☐
n	2

LABELS

A Year	B i (%)	C Date	D Deposit	E Withdraw
a	b	c	d	e
0 USED	1	2	3	4
5	6	7	8	9 Jan.-Feb.

REGISTERS

0 Tot. int.	1 Bal.	2 N = days	3 Year	4 Date	5 USED	6 36525	7	8	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

Program Description I

Program Title ACCUMULATED INTEREST/REMAINING BALANCE

Contributor's Name HEWLETT-PACKARD COMPANY
Address Corvallis Division
1000 N.E. Circle Boulevard
City Corvallis, OR 97330 **State** **Zip Code**

Program Descripti

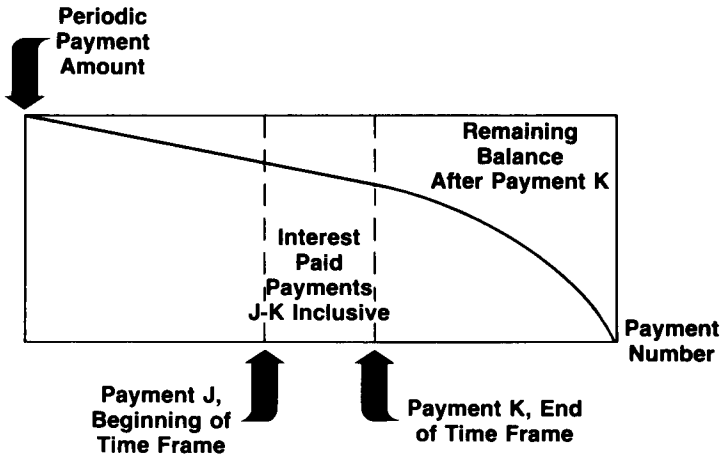


Figure 4

This program finds both the total interest paid over a specified number of payment periods and the remaining balance at the end of the last specified period, given the periodic interest rate, periodic payment amount, loan amount, and the beginning and ending payment numbers for the time span being considered. The payments associated with both the beginning (J) and the ending (K) payment period are included in the calculation.

The program can be used for loans with a balloon payment as well as loans arranged to be fully amortized provided two cautions are observed. First, the balloon payment of the loan must be at the same time as, and in addition to the last payment. Second, care should be taken not to enter a value for K that is after the last payment since the program has no way of knowing the term of the loan.

Operating Limits a

An option is available to output the amortization schedule between payments J and K (f A).

Pressing f E sets and clears the print flag. Successive use of f E will alternately display 1.00 and 0.00, indicating that the print/pause mode is on or off respectively.

The data generated is valid for loans that have a balloon payment, as well as those that are arranged to be fully amortized. For loans with a balloon payment, the remaining balance of the last payment period is the balloon payment due in addition to the last periodic payment.

Program Description I

Program Title

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables

For loans scheduled to be fully amortized, the remaining balance after the last payment period may be slightly more or less than zero. This is because the program assumes that **all** payments are equal to the value entered for PMT. In fact for most loans, the last payment is slightly more or less than the rest.

The calculator performs all internal calculations to ten digits. If the user wishes to round the schedule to dollars and cents, the following sequence may be used:

1. Press GTO .113
2. Switch to PRGM mode.
3. Press RND
4. Switch back to RUN mode.

Operating Limits and Warnings

Program Description II

Sketch(es)

Sample Problem(s)

5. Accumulated Interest/Remaining Balance

$$BAL_K = \frac{1}{(1 + i)^{-K}} \left[PMT \frac{(1 + i)^{-K} - 1}{i} + PV \right]$$

$$Int_{J-K} = BAL_K - BAL_{J-1} + (K - J + 1) \cdot PMT$$

where:

k^{th} payment to principal $= BAL_{K-1} - BAL_K$
k^{th} payment to interest $= PMT - (BAL_{K-1} - BAL_K)$
Total payment to interest $= (K) \times (PMT) - (PV - BAL_K)$

Solution(s)

Reference(s)

Program Description II

Sketch(es)

Example 1:

A mortgage is arranged such that the first payment is made at the end of October, 1975 (i.e., October is payment period 1). It is a $20,000 loan at 9%, with monthly payments of $167.84. What is the accumulated interest for 1975 (periods 1-3) and 1976 (periods 4-15) and what would the remaining balance be at the end of each year?

Sample Problem(s)

Keystrokes:	Outputs:	
1 A 3 A 9 ENTER↑ 12 ÷ B		
167.84 C 20000 D E →	449.60	(interest paid in 1975)
R/S →	19946.08	(remaining balance at the end of 1975)
4 A 15 A E →	1785.89	(interest paid in 1976)
R/S →	19717.88	(remaining balance at the end of 1976)

Example 2:

Generate an amortization schedule for the first two payments of a $30,000, 7% mortgage having monthly payments of $200. Then jump ahead and generate the data for the 36th payment.

Keystrokes:	Outputs:	
1 A 2 A 7 ENTER↑ 12 ÷ B		
200 C 30000 D f A →	1.00	(starting 1st period)
R/S →	175.00	(payment to interest)
R/S →	25.00	(payment to principal)
R/S →	29975.00	(remaining balance)
R/S →	175.00	(total interest to date)
R/S →	2.00	(starting 2nd period)
R/S →	174.85	(payment to interest)
R/S →	25.15	(payment to principal)
R/S →	29949.85	(remaining balance)
R/S →	349.85	(total interest to date)

Solution(s)

Keystrokes:	Outputs:	

Now let's skip ahead to the 36th payment period.

36 A A f A →	36.00	(starting 36th period)
R/S →	169.36	(payment to interest)
R/S →	30.64	(payment to principal)
R/S →	29001.75	(remaining balance)
R/S →	6201.75	(total interest to date)

Reference(s)

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load side 1 and side 2.			
2	Optional: Select print/pause			
	mode for amortization			
	schedule.		f E	1.00 or 0.00
3	Key in			
	• Starting period number	J	A	J
	• Ending period number	K	A	K
	• Periodic interest rate	i (%)	B	i (%)
	• Periodic payment amount	PMT	C	PMT
	• Initial loan amount	PV	D	PV
4	Compute the total interest paid			
	between periods J and K			
	inclusive, and the remaining			
	balance at the end of period K.		E	INT
			R/S	BAL
	OR			
5	Generate the amortization			
	schedule between payments J			
	and K inclusive. If the			
	print/pause mode is on (1.00),			
	the results are printed			
	automatically.		f A	J
6	Calculate amount paid to			
	interest for period J.		R/S	PMT to INT
7	Calculate amount paid to			
	principal for period J.		R/S	PMT to PRIN
8	Calculate remaining balance at			
	the end of period J.		R/S	BAL
9	Calculate total interest paid			
	between periods J thru K			
	inclusive.		R/S	TOT INT
10	Increment J for next period.		R/S	J + 1
	If J ≤ K, go to step 6 for next			
	period's values. Otherwise,			
	stop.			
11	For a new case, go to step 2 and			
	change appropriate input			
	values.			

STEP	UT UNITS	KEYS	OUTPUT DATA/UNITS

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBLA	21 11	
002	RCL0	36 00	
003	STO7	35 07	$J \rightarrow R_7$
004	X≷Y	-41	
005	STO0	35 00	$K \rightarrow R_0$
006	RTN	24	
007	*LBLB	21 12	
008	EEX	-23	
009	2	02	
010	÷	-24	
011	STO1	35 01	$i/100 \rightarrow R_1$
012	LSTX	16-63	
013	×	-35	
014	RTN	24	
015	*LBLC	21 13	
016	STO2	35 02	$PMT \rightarrow R_2$
017	RTN	24	
018	*LBLD	21 14	
019	STO3	35 03	$PV \rightarrow R_3$
020	RTN	24	
021	*LBLE	21 15	
022	RCL0	36 00	
023	RCL7	36 07	
024	X≤Y?	16-35	
025	GTO0	22 00	
026	STO0	35 00	
027	R↓	-31	
028	STO7	35 07	
029	*LBL0	21 00	
030	1	01	
031	RCL1	36 01	
032	+	-55	$(1 + i/100) \rightarrow R_8$
033	STO8	35 08	
034	RCL0	36 00	
035	GSB1	23 01	$BAL_K \rightarrow R_4$
036	STO4	35 04	
037	RCL8	36 08	
038	RCL7	36 07	
039	1	01	
040	-	-45	
041	GSB1	23 01	$-BAL_{J-1}$
042	CHS	-22	
043	RCL4	36 04	$BAL_K - BAL_{J-1} \rightarrow R_6$
044	+	-55	
045	STO6	35 06	
046	RCL0	36 00	
047	RCL7	36 07	
048	-	-45	
049	1	01	
050	+	-55	
051	RCL2	36 02	
052	×	-35	
053	+	-55	INT_{J-K}
054	RTN	24	BAL_K
055	RCL4	36 04	
056	R/S	51	

STEP	KEY ENTRY	KEY CODE	COMMENTS
057	*LBLa	21 16 11	
058	RCL7	36 07	
059	F0?	16 23 00	
060	SPC	16-11	
061	GSB9	23 09	
062	1	01	
063	RCL1	36 01	
064	+	-55	
065	STO8	35 08	
066	RCL7	36 07	
067	GSB1	23 01	
068	STO4	35 04	
069	RCL8	36 08	
070	RCL7	36 07	
071	1	01	
072	-	-45	
073	GSB1	23 01	
074	RCL4	36 04	
075	-	-45	
076	STO6	35 06	
077	RCL2	36 02	
078	X≷Y	-41	
079	-	-45	INT_J
080	GSB9	23 09	
081	RCL6	36 06	
082	GSB9	23 09	$PRINC_J$
083	RCL4	36 04	
084	GSB9	23 09	$RBAL_J$
085	RCL7	36 07	
086	RCL2	36 02	
087	×	-35	
088	RCL3	36 03	
089	RCL4	36 04	
090	-	-45	
091	-	-45	
092	GSB9	23 09	TOT INT
093	1	01	
094	ST+7	35-55 07	
095	RCL0	36 00	
096	RCL7	36 07	
097	X≤Y?	16-35	$J \leq K$?
098	GTOa	22 16 11	
099	RTN	24	
100	*LBL1	21 01	
101	CHS	-22	
102	Y^x	31	
103	STO5	35 05	
104	1	01	
105	-	-45	
106	RCL1	36 01	
107	÷	-24	
108	RCL2	36 02	
109	×	-35	
110	RCL3	36 03	
111	+	-55	
112	RCL5	36 05	

REGISTERS

0	1	2	3	4	5	6	7	8	9
K	i/100	PMT	PV	Used	Used	Used	J	1+i/100	
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
113	÷	-24	RND may be inserted here.
114	RTN	24	
115	*LBLe	21 16 15	
116	F0?	16 23 00	
117	GTO2	22 02	
118	SF0	16 21 00	
119	1	01	
120	RTN	24	
121	*LBL2	21 02	
122	0	00	
123	CF0	16 22 00	
124	RTN	24	
125	*LBL9	21 09	
126	F0?	16 23 00	
127	GTO3	22 03	
128	R/S	51	
129	RTN	24	
130	*LBL3	21 03	
131	PRTX	-14	
132	RTN	24	

Print mode option.

STEP	KEY ENTRY	KEY CODE	COMMENTS
170			
180			
190			
200			
210			
220			

LABELS

A J,K	B i	C PMT	D PV	E INT;RB	0 PRINT?
a SKD	b	c	d	e PRINT?	1
0 Used	1 Used	2	3	4	2
5	6	7	8	9	3

SET STATUS

FLAGS		TRIG		DISP	
	ON OFF				
0	☐ ☒	DEG	☒	FIX	☒
1	☐ ☒	GRAD	☐	SCI	☐
2	☐ ☒	RAD	☐	ENG	☐
3	☐ ☒			n	2

Program Description I

Program Title Stock Portfolio Valuation

Contributor's Name Hewlett-Packard

Address

City **State** **Zip Code**

Program Description, Equations, Variables Data cards created with a separate data program provide initial purchase price of a stock and the number of shares for a portfolio of any size. Valuation program prompts user one stock at a time. User inputs current market price and annual dividend. Price input: 25-5/8 is inputed as 25.58. Program returns the percent change of value of each stock and prompts the user for the next stock. If more than one data card is used the program prompts user by flashing repeditive 18's until a new data card is inserted.

When all current prices have been entered, user initiates the valuation of the total portfolio. Output includes original portfolio value, new portfolio value, % change in value, date original portfolio was created, and annual dividend yield as a percent of current market value.

Operating Limits and Warnings Shares selling for more than 999 dollars @ can not be used (such shares have existed although rare).

Program Description II

Sketch(es)

Sample Problem(s) Sample data includes the following information:
1) 100 shares at 25-5/8 @, 2) 200 at 30-1/4, 3) 50 at 89-7/8, 4) deleted stock
5) 500 at 65-1/4 [data is packed by data program so that register 1 contains
100.025625] Date portfolio created 10.25 1977.

Current information: 1) $27-1/4 with $1.70 dividend 2) 33-1/2 with 2.10
3) 96-1/8 with 4.55 4) none 5) 64-3/8 with 3.50

Solution(s)

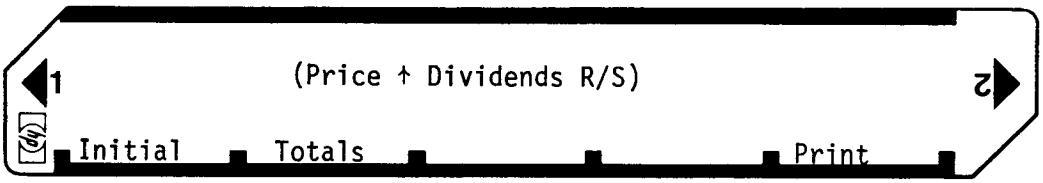
Prompt	Input A	Output	Input
1	27.14 [↑] 1.7 [R/S]	6.34	[R/S]*
2	33.12 [↑] 2.1 [R/S]	10.74	[R/S]*
3	96.18 [↑] 4.55[R/S]	6.95	[R/S]*
4	(immediately outputs a zero)	0	[R/S]
5	64.38 [↑] 3.50[R/S]	-1.34	[B]
	Original value	45731.25	[R/S]*
	New value	46418.75	[R/S]*

Reference(s)

% change in value	1.5	[R/S]*
total yearly dividend	2567.50	[R/S]*
yearly dividend yield	5.53	[R/S]*
date portfolio created	10.25 1977	

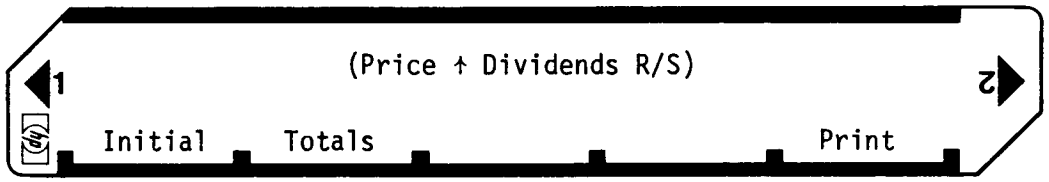
* Only necessary if print option not exercised.

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Clear register: This procedure is not necessary if the calculator has just been switched on.		CL REG	
			P≤S	
			CL REG	
2.	Load side 1 and 2 of program			
3.	Load 1st data card			
4.	Select print option (97)		E	1
	Alternate presses of [E] sets (1) and unsets (0) the print option			
5.	Initialize		A	1
6.	Key in current stock price			
	27-1/4 would enter as 27.14	27.14	ENTER	27.14
7.	Key in annual dividend	1.7	R/S	
	Output is % change in this stock			6.34
8.	Proceed with steps 6-8 until all prices are entered.		R/S*	2
	If a stock has been deleted (register is filled with zeros) the program displays zero immediately. Continue by pressing R/S.			
	If there are additional data cards (18 stocks per card), the last entry will flash 18 until a new card is entered.			
	* Not necessary if print option has been selected.			
	Continued on next page --------->			

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
	After all data is entered: Old portfolio		B	$
	total.			
	New portfolio total		R/S*	$
	Change in portfolio value from purchase		R/S*	%
	Total annual dividends		R/S*	$
	Portfolio dividend yield as a percent of current market value.		R/S*	$
	Date original portfolio created		R/S*	MM.DDYYYY
	* Not necessary if print option has been			
	selected.			

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBLA	21 11	Set I to 0 for portfolio register run
002	DSP2	-63 02	
003	1	01	
004	STOI	35 46	
005	*LBL1	21 01	Recall ith historical stock data & check for a deleted stock
006	RCLi	36 45	
007	X=0?	16-43	
008	GTO7	22 07	
009	INT	16 34	
010	STOE	35 15	Unpack data store # of shares in E, calculate and store original stock value in D, and accumulate original portfolio value in A
011	LSTX	16-63	
012	FRC	16 44	
013	EEX	-23	
014	3	03	
015	x	-35	
016	x	-35	
017	STOD	35 14	
018	RCLA	36 11	
019	+	-55	
020	STOA	35 11	
021	RCLI	36 46	Prompt for current input
022	RCL0	36 00	
023	+	-55	
024	F0?	16 23 00	
025	PRTX	-14	
026	R/S	51	
027	RCLE	36 15	Accumulate total dividend in C
028	x	-35	
029	RCLC	36 13	
030	+	-55	
031	STOC	35 13	
032	R↓	-31	
033	GSBa	23 16 11	Normalize price
034	RCLE	36 15	
035	x	-35	
036	RCLB	36 12	Accumulate current value in B
037	X≷Y	-41	
038	+	-55	
039	STOB	35 12	
040	LSTX	16-63	Calculate and display % change in ith stocks value
041	RCLD	36 14	
042	X≷Y	-41	
043	%CH	16 55	
044	GSB5	23 05	print option
045	*LBL2	21 02	
046	ISZI	16 26 46	Check for end of registers. If less than 18 continue else:
047	1	01	
048	8	08	
049	RCLI	36 46	
050	X≤Y?	16-35	
051	GTO1	22 01	
052	CF3	16 22 03	
053	X≷Y	-41	
054	STOI	35 46	Data card merge and prompt
055	RCL0	36 00	
056	+	-55	
057	*LBLc	21 16 13	
058	MRG	16-62	
059	PSE	16 51	
060	F3?	16 23 03	
061	GTOA	22 11	Continue prompting with i+18 & loading
062	GTOc	22 16 13	
063	*LBLB	21 12	
064	RCLA	36 11	Original portfolio value
065	GSB5	23 05	
066	RCLB	36 12	New portfolio value
067	GSB5	23 05	
068	%CH	16 55	
069	GSB5	23 05	Change in value
070	DSP2	-63 02	
071	RCLC	36 13	Total yearly dividend
072	GSB5	23 05	
073	LSTX	16-63	
074	÷	-24	Total dividend
075	1	01	
076	0	00	yield as a % of
077	0	00	current value
078	x	-35	
079	GSB5	23 05	
080	P≷S	16-51	
081	RCL9	36 09	Date portfolio
082	P≷S	16-51	Created
083	DSP6	-63 06	
084	GSB5	23 05	
085	DSP2	-63 02	
086	R/S	51	
087	*LBLE	21 15	
088	F0?	16 23 00	Print/no print
089	GTO4	22 04	
090	SF0	16 21 00	flag set
091	1	01	
092	RTN	24	
093	*LBL4	21 04	
094	0	00	
095	CF0	16 22 00	
096	RTN	24	
097	*LBL5	21 05	
098	F0?	16 23 00	Print or no
099	GTO6	22 06	
100	R/S	51	print decision
101	RTN	24	
102	*LBL6	21 06	
103	PRTX	-14	
104	SPC	16-11	
105	RTN	24	Print
106	R/S	51	
107	*LBL7	21 07	
108	RCLI	36 46	
109	RCL0	36 00	Display contents
110	+	-55	of a 0 register
111	PSE	16 51	

REGISTERS

0 Mult Crd	1	2	3	4 --STOCKS	5	6	7	8	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9 Date
A Old Port Total	B New Port Total		C Total Div	D Old Stock Val.		E # of Shrs.			I Used

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
112	X≷Y	-41	
113	R/S	51	
114	GTO2	22 02	Return to sequential review
115	*LBLa	21 16 11	
116	ENT↑	-21	
117	FRC	16 44	
118	X=0?	16-43	
119	GTOb	22 16 12	
120	EEX	-23	
121	1	01	
122	x	-35	
123	INT	16 34	Convert CC.DN into CC + D/N checking to avoid division by 0.
124	LSTX	16-63	
125	FRC	16 44	
126	÷	-24	
127	EEX	-23	
128	1	01	
129	÷	-24	
130	X≷Y	-41	
131	INT	16 34	
132	*LBLb	21 16 12	
133	+	-55	
134	RTN	24	
135	R/S	51	

STEP	KEY ENTRY	KEY CODE	COMMENTS
140			
150			
160			
170			
180			
190			
200			
210			
220			

LABELS

A Initil	B Summary	C	D	E Print?	
a Fract Dec	b	c	d	e	
0	1 Unpack	2 End Check	3 Crd Prmpt	4 Print	
5 Print	6 Print	7	8	9	

FLAGS

0 Print?
1
2
3 Merge

SET STATUS

FLAGS	ON	OFF
0	☒	☐
1	☒	☐
2	☒	☐
3	☒	☐

TRIG		DISP	
DEG	☐	FIX	☒
GRAD	☐	SCI	☐
RAD	☐	ENG	☐
		n	2

Program Description I

Program Title Portfolio Data Card

Contributor's Name Hewlett-Packard

Address 1000 N.E. Circle Blvd.

City Corvallis **State** Oregon **Zip Code** 97330

Program Description, Equations, Variables Registers 1 through 18 are used to store historic cost and quantity data on individual stocks. Each register represents one stock. If N represents number of shares, C represents the integer dollar cost and F the fractional cost, the register is packed as NNNN.CCCFFF*. Program sequentially prompts user for input. Number of shares and price are entered. User can load prices with fractions: 25-7/8 is 25.78.

Options include deleting stocks (filling a register with 0's), adding stock, and correcting erroneous entries. Register 0 contains the date the portfolio was assembled. MM.DDYYYY.

Operating Limits and Warnings *CCC is limited to three digits. Fractions are limited to single digit denominators.

Program Description II

Sketch(es)

Sample Problem(s) User has a portfolio of 5 stocks which was purchased on October 25, 1977. Stocks are as follows:

Stock No.	No. of Shares	Price
1	100	25-5/8
2	400	66
3	50	89-7/8
4	300	18-3/8
5	500	65-1/4

At a later date stock 2 and 4 are sold. Subsequent to that a new stock is purchased.

	200	30-1/4

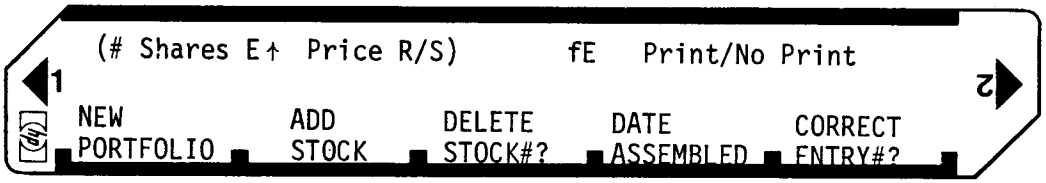
Solution(s)

Prompts	Input	Output
	10.25 1977 [D]	10.25 1977
	[A]	
1	100 [↑] 25.58 [R/S]	
2	400 [↑] 66 [R/S]	
3	50 [↑] 89.78 [R/S]	
4	300 [↑] 18.38 [R/S]	
5	500 [↑] 65.14 [R/S]	
6 (ignore)	[Write Data] or [W/Data]	

Reference(s)

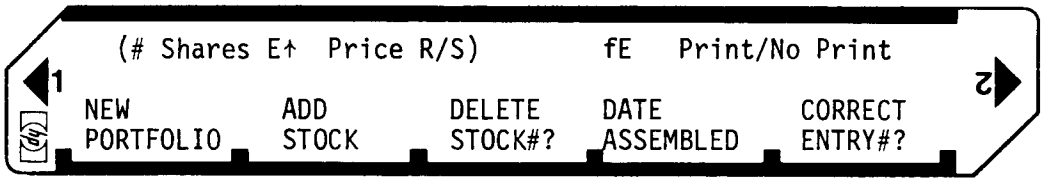
Later	(Enter program and data cards)	
	2 [C]	400.06600 [R/S] 0
	4 [C]	300.018375[R/S] 0
	[B]	
	200 [↑] 30.14 [R/S]	200.030250
	[Write Data] or [W/Data]	

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Clear registers - this is not necessary if calculator has just been switched on		CL REG P<S> CL REG	
2.	Enter program card			
3.	If you have 97 and wish to have a printed record		f E	1
4.	If portfolio is being created, key in the assembly or purchase date	MM.DDYYYY	D	
5.	If portfolio is being created		A	1
	a) Number of shares		ENTER↑	
	b) Price(25 5/8 keyed in as 25.58)		R/S	Next Reg#
	Repeat a & b until all stocks are entered			
	If more than 18 stocks are being entered, the program will automatically prompt for a data (blank) card after the 18th entry. After the card has been entered a 0 appears. Press [R/S] to continue		R/S	CRD 0 19
	Complete all stock entries before returning to make any corrections. If more than 1 data card is required, re-enter the appropriate card after the all the stocks have been entered If only one card is used, corrections (etc.) can be done after the last stock is entered.			
	Continued on next page ---------->			

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
	OPTIONS			
B	ADD STOCK - To a previously created data card			
	enter program and data cards and press →		B	
	If space is available, the program will			
	prompt user with the register #. Use			
	5 a & b, then back to B for additional			
	available registers.			
C	DELETE STOCK - Key in register # and press	REG #	C	
	Program displays current register contents.			
	If user desires to delete press →		R/S	
	Or else go back to B,C, or E.			
E	CORRECT ENTRY # - Key in register # of reg to			
	be corrected and press →	REG #	E	Reg #
	then 3 a & b.			
	PRINT/NO PRINT - 1 represents print.		f e	
	Alternate presses of [f] [E] sets and			
	unsets the print option.			

97 Program Listing 1

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBLA	21 11	
002	0	00	Initialize
003	STOI	35 46	
004	*LBL1	21 01	
005	ISZI	16 26 46	Sequential data
006	GSB3	23 03	entry
007	GSBa	23 16 11	
008	GTO1	22 01	
009	RTN	24	
010	*LBLB	21 12	
011	1	01	
012	STOI	35 46	Initialize
013	*LBL2	21 02	sequential register
014	GSB3	23 03	search for first
015	RCLi	36 45	zero register
016	X=0?	16-43	
017	GTOa	22 16 11	
018	ISZI	16 26 46	
019	GTO2	22 02	
020	RTN	24	
021	*LBLC	21 13	
022	DSP6	-63 06	
023	RCL0	36 00	Recall register to
024	-	-45	be deleted and
025	STOI	35 46	display
026	RCLi	36 45	
027	R/S	51	
028	0	00	
029	STOi	35 45	Delete (store 0)
030	DSP2	-63 02	designed register
031	RTN	24	
032	*LBLD	21 14	
033	DSP6	-63 06	
034	F0?	16 23 00	Store date in
035	PRTX	-14	register 19
036	P↕S	16-51	
037	STO9	35 09	
038	P↕S	16-51	
039	DSP2	-63 02	
040	F0?	16 23 00	
041	SPC	16-11	
042	RTN	24	
043	*LBLE	21 15	
044	DSP6	-63 06	
045	RCL0	36 00	Set I register to
046	-	-45	
047	STOI	35 46	store change
048	GSBa	23 16 11	
049	DSP2	-63 02	
050	R/S	51	
051	*LBLa	21 16 11	
052	RCLI	36 46	Prompt user with
053	RCL0	36 00	register #
054	+	-55	
055	GSB5	23 05	
056	X↕Y	-41	
057	F0?	16 23 00	Print inputed
058	PRTX	-14	
059	X↕Y	-41	Data
060	F0?	16 23 00	
061	PRTX	-14	
062	GSBb	23 16 12	Normalize price
063	EEX	-23	
064	3	03	
065	÷	-24	
066	+	-55	Pack register
067	F0?	16 23 00	
068	SPC	16-11	NNN.CCCFFF
069	F0?	16 23 00	
070	SPC	16-11	
071	STOi	35 45	
072	RTN	24	
073	*LBL3	21 03	
074	1	01	
075	8	08	Check for end of
076	RCLI	36 46	registers
077	X≤Y?	16-35	
078	RTN	24	
079	R↓	-31	
080	WDTA	16-61	Output data
081	0	00	
082	R/S	51	
083	R↓	-31	
084	RCL0	36 00	Clear registers
085	+	-55	and begin loading
086	CLRG	16-53	data for subsequent
087	P↕S	16-51	card
088	CLRG	16-53	
089	STO0	35 00	
090	GTOA	22 11	
091	RTN	24	
092	*LBLb	21 16 12	
093	ENT↑	-21	
094	FRC	16 44	Normalize Price
095	X=0?	16-43	
096	GTOc	22 16 13	CCC.ND
097	EEX	-23	becomes
098	1	01	CCC + $\frac{N}{D}$
099	x	-35	
100	INT	16 34	(Avoids ND = 0)
101	LSTX	16-63	
102	FRC	16 44	
103	÷	-24	
104	EEX	-23	
105	1	01	
106	÷	-24	
107	X↕Y	-41	
108	INT	16 34	
109	*LBLc	21 16 13	
110	+	-55	

REGISTERS

0 DATE	1	2	3	4	5 STOCKS	6	7	8	9
S0	S1	S2	S3	S4	S5 STOCKS	S6	S7	S8	S9
A	B	C			D	E		I USED	

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
111	RTN	24	
112	*LBLe	21 16 15	
113	F0?	16 23 00	
114	GTO4	22 04	
115	SF0	16 21 00	Print/No Print
116	1	01	SET
117	RTN	24	
118	*LBL4	21 04	
119	0	00	
120	CF0	16 22 00	
121	RTN	24	
122	*LBL5	21 05	
123	F0?	16 23 00	
124	GTO6	22 06	
125	R/S	51	Operationalize
126	RTN	24	print/no print
127	*LBL6	21 06	
128	PRTX	-14	
129	R/S	51	
130	RTN	24	
131	R/S	51	
140			
150			
160			
170			
180			
190			
200			
210			
220			

LABELS					FLAGS	SET STATUS		
A Start	B Add	C Delete	D Date	E Correct	0 Print/ No Print	FLAGS	TRIG	DISP
a Used	b Normalize Price	c Used	d	e Print/ No Print	1 Additional data card	ON OFF 0 ☐ ☒ 1 ☐ ☒ 2 ☐ ☒ 3 ☐ ☒	DEG ☐ GRAD ☐ RAD ☐	FIX ☒ SCI ☐ ENG ☐ n 2
0	1 Data Entry	2 Add Search Routine	3 Last Reg Check	4 Used	2			
5 Used	6 Used	7	8	9	3			

Program Description I

Program Title TRUE ANNUAL GROWTH RATE (DCF) OF INVESTMENT PORTFOLIO

Contributor's Name Ken L. Singer
Address 2323 Augusta Drive
City Houston **State** Texas **Zip Code** 77057

Program Description, Equations, Variables, etc. This program finds the true annual growth rate (discounted cash flow rate of return) of an investment portfolio or any unlimited cash flow stream. Inputs are as follows:

1. Evaluation date and market value
2. Lump-sum payments and/or withdrawals: date and amount
3. Series payments and/or withdrawals: starting date of series; number of payments or withdrawals in series; months between each payment or withdrawal; and amount of each payment or withdrawal

(The program can be used to find the DCF rate of return of a standard cash flow stream by treating cash flow investment outlays the same as portfolio withdrawals and cash flow revenues the same as portfolio payments; the date and amount of the initial cash flow investment is input as the portfolio evaluation date and market value.) For an investment portfolio, a dividend which is not reinvested is treated as a withdrawal. For a cash flow stream, a continuous flow can be approximated by many small series payments. For example, $1000 received continuously over a year can be approximated by 100 revenues, received

Operating Limits and Warnings (1) Total payments cannot equal total withdrawals (including market value), i.e. zero growth rate. (2) As in any discounted cash flow analysis, if the year by year cumulative net cash flow (payments minus withdrawals) changes sign more than once, there may not be a unique rate of return. Such a case will be indicated by widely differing values of i, i_1, and i_2; accordingly, the final rate will be incorrect. (3) The growth rate must be algebraically greater

Program Description I

Program Title TRUE ANNUAL GROWTH RATE (DCF) OF INVESTMENT PORTFOLIO

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables, etc. (cont'd)

at intervals of 12/100 months, in the amount of $1000/100 each.

Input data are entered three times. Program determines its own initial guess for the rate from the first entry of the data (Pass 1). The initial rate is then used to discount the payments/withdrawals input in the second entry, and the resulting ratio of total discounted withdrawals to total discounted payments is used to calculate a refined rate (Pass 2). In the same manner data are entered a third time and discounted using the refined rate to obtain a further refined rate; then the initial, refined, and further refined rates are combined to obtain a final rate (Pass 3). Accuracy averages 99.999 %.

Let: i = initial rate, % $m = 1 + (i/100)$

i_1 = refined rate, % $m_1 = 1 + (i_1/100)$

i_2 = further refined rate, % $m_2 = 1 + (i_2/100)$

i_c = final rate, %

L = "lump-sum" S = "series"

W = withdrawal amount P = payment amount

TW = total withdrawals TP = total payments

Operating Limits and Warnings (cont'd)

than -100 percent. (4) In some other rare instances a particular set of data could cause division by zero. If this instance should occur, it is suggested that the market value (portfolio) or initial investment outlay (cash flow) be changed by a very small amount; the entire program should then be rerun.

Program Description I

Program Title TRUE ANNUAL GROWTH RATE (DCF) OF INVESTMENT PORTFOLIO

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables, etc. (cont'd)

TDW = total discounted wdls. TDP = total discounted pmts.

n = time (years) of payment/withdrawal (relative to evaluation date)

n' = time (years) series payment/withdrawal starts

I = interval (months) between series pmt./wdl. I' = I/12

N = number of series payments/withdrawals

Pass 1: $\bar{n} = n' + (I'N - I)/2$

$$x_w = \frac{1}{TW}\left[\sum(W_L)(n) + \sum(W_S)(N)(\bar{n})\right]$$

$$x_p = \frac{1}{TP}\left[\sum(P_L)(n) + \sum(P_S)(N)(\bar{n})\right]$$

$$a = x_w - x_p \qquad m = (TW/TP)^{\frac{1}{a}}$$

Operating Limits and Warnings

Program Description I

Program Title TRUE ANNUAL GROWTH RATE (DCF) OF INVESTMENT PORTFOLIO

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables, etc. (cont'd)

Pass 2:

$$TDW = \sum(W_L)(m^{-n}) + \sum(W_S)\left(\frac{m^{I'N}-1}{m^{I'}-1}\right)(m^{-(n' + I'N - I')})$$

$$TDP = \sum(P_L)(m^{-n}) + \sum(P_S)\left(\frac{m^{I'N}-1}{m^{I'}-1}\right)(m^{-(n' + I'N - I')})$$

$$1/b = (\log \frac{TW}{TP})/(\log \frac{TW/TP}{TDP/TDW}) \qquad m_1 = m^{\frac{1}{b}}$$

Pass 3:

TDW_1 = same as Pass 2, except m_1 used instead of m

TDP_1 = same as Pass 2, except m_1 used instead of m

$$1/b_1 = (\log \frac{TW}{TP})/(\log \frac{TW/TP}{TDP_1/TDW_1}) \qquad m_2 = m_1^{\frac{1}{b_1}}$$

$$i_c = \left(m + \frac{(m_1 - m)^2}{2m_1 - m - m_2} - 1\right)(100)$$

Operating Limits and Warnings

Program Description II

Sketch(es)

Sample Problem(s)

(1) Investment portfolio: $2600 and $3600 were paid into a fund on 3/1/67 and 5/1/70 respectively. $2000 was withdrawn on 4/1/73. Five quarterly dividends of $60 each were paid by the fund (and not reinvested) starting 11/1/68. Additionally, twelve monthly amounts of $100 each were invested in the fund beginning 2/1/74. What was the true annual growth rate of the fund as of 4/1/76, when it had a value of $7000?

Solution(s)

```
E  4.1976 ↑ 7000 A
   3.1967 ↑ 2600 B
   5.1970 ↑ 3600 B
   2.1974 ↑ 12 ↑ 1 ↑ 100 C
   4.1973 ↑ 2000 CHS B
  11.1968 ↑  5 ↑ 3 ↑ 60 CHS C D ──→ 1.0425(965)        (Pass 1)
                                 ──→ 1.0420(671)        (Pass 2)
                                 ──→ 1.0420(730)
                                     4.2072(893) pct.   (Pass 3)
```

Reference(s)

Program Description II

Sketch(es)

Sample Problem(s)

(2) Cash flow: (All figures, except those with asterisk, are lump-sum at end of year)

Year	0	1	2	3	4	5	6	7	8
Investment	5	0	8	8	8	7	0	0	0
Revenue	0	3	4	4	4	4	4	9*	9*

*continuously received from start of year to end of year

Solution(s)

```
E  0.0000 ↑ 5 A
   0.0001 ↑ 3 B
   0.0002 ↑  5 ↑ 12 ↑  4 C
   0.0006 ↑ 200 ↑.12 ↑.09 C
   0.0005 ↑ 7 CHS B
   0.0002 ↑  3 ↑ 12 ↑  8 CHS C D→1.0631(700)      (Pass 1)
                                →1.0652(102)      (Pass 2)
                                →1.0652(778)
                                  6.5280(152) pct.(Pass 3)
```

Reference(s) (1) "Changing Times computer service: Find out how your investments are really doing", Changing Times Magazine, March 1970, pgs. 47-49; (2) Wild, N. H., "Return on Investment made easy", Chemical Engineering Magazine, April 12, 1976, pgs. 153-154

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load sides 1 and 2			
2	Press E until 1.0000 is displayed		E	1 (Pass)
3	Initialize data entry sequence as follows:			
	Enter evaluation date for investment portfolio or date of initial capital investment for cash flow	MM.YYYY	↑	
	Enter market value of portfolio or initial investment amount of cash flow	Amount	A	Amount
4	If there are any lump-sum payments (for portfolio) or revenues (for cash flow), input them as follows:			
	Enter date	MM.YYYY	↑	
	Enter amount	Amnt (A)	B	-(A)(n)
	(Repeat step 4 as necessary)			
5	If there are any series payments (for portfolio) or revenues (for cash flow), input them as follows:			
	Enter starting date of series	MM.YYYY	↑	
	Enter number of payments in series	N	↑	
	Enter interval (months) between payments	Months	↑	
	Enter amount of each payment	Amnt (A)	C	$-AN\overline{n}$
	(Repeat step 5 as necessary)			
6	If there are any lump-sum withdrawals (for portfolio) or investment outlays (for cash flow), input them as follows:			
	Enter date	MM.YYYY	↑	
	Enter amount	Amnt (A)	CHS B	(A)(n)
	(Repeat step 6 as necessary)			
7	If there are any series withdrawals (for portfolio) or investment outlays (for cash flow), input them as follows:			
	Enter starting date of series	MM.YYYY	↑	
	Enter number of withdrawals in series	N	↑	
	Enter interval (months) between withdrawals	Months	↑	
	Enter amount of each withdrawal	Amnt (A)	CHS C	$(A)(N)(\overline{n})$
	(Repeat step 7 as necessary)			

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
8	To calculate initial rate, press		D	1+i/100
9	Press E until 2.0000 is displayed		E	2 (Pass)
10	Repeat step 3			-Amount
11	Repeat step 4			P.Value
12	Repeat step 5			P.Value
13	Repeat step 6			P.Value
14	Repeat step 7			P.Value
15	To calculate refined rate, press		D	$1+i_1/100$
16	Press E until 3.0000 is displayed		E	3 (Pass)
17	Repeat step 3			-Amount
18	Repeat step 4			$P.Value_1$
19	Repeat step 5			$P.Value_1$
20	Repeat step 6			$P.Value_1$
21	Repeat step 7			$P.Value_1$
22	To calculate further refined and final rates, press		D	$1+i_2/100$***
				i_e (%)
	(If a mistake is made during data entry and noticed before completing the step, it is only necessary to start that step over again.)			
	(If a mistake is made during data entry and noticed after completing a step, it is only necessary to go back to the step immediately following the "Press E until ..." at the start of the pass in which the error was made.)			
	(For another problem, repeat steps 2-22.)			
	*** pause on HP-67; print on HP-97			
	P.Value = Present (discounted) Value			

EVALUATION DATE AND MARKET VALUE (PORTFOLIO)
OR DATE AND AMOUNT OF INITIAL INVESTMENT OUTLAY (CASH FLOW)

DATE AMOUNT

________ [↑] ________ [A]

LUMP-SUM PAYMENTS (PORTFOLIO) OR REVENUES (CASH FLOW)

DATE AMOUNT

________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]
________ [↑] ________ [B]

SERIES PAYMENTS (PORTFOLIO) OR REVENUES (CASH FLOW)

ST. DATE NUMBER INTERVAL AMOUNT

________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]
________ [↑] ________ [↑] ________ [↑] ________ [C]

LUMP-SUM WITHDRAWALS (PORTFOLIO) OR INVESTMENT OUTLAYS (CASH FLOW)

DATE AMOUNT

________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]
________ [↑] ________ [CHS] [B]

SERIES WITHDRAWALS (PORTFOLIO) OR INVESTMENT OUTLAYS (CASH FLOW)

ST. DATE NUMBER INTERVAL AMOUNT

________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]
________ [↑] ________ [↑] ________ [↑] ________ [CHS] [C]

67 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	g LBL fe	32 25 15	Convert date to -n
	h X⇄Y	35 52	Amount in y; Date in X
	↑	41	MM.YYYY in X & Y Regs.
	g FRAC	32 83	.YYYY in X Reg.
	−	51	MM
	h LST X	35 82	Bring back .YYYY
	EEX	43	YYYY
	4	04	
	X	71	
010	h X⇄Y	35 52	MM. in X; YYYY. in Y
	1	01	fraction of year = MM/12
	2	02	
	÷	81	
	+	61	YYYY. fraction
	RCL 5	34 05	Eval. Date (0 in Pass 1)
	h X⇄Y	35 52	-n = (Eval. Date) − Date
	−	51	
	h RTN	35 22	
	f LBL A	31 25 11	Initialize
020	h F? 0	35 71 00	Test for Pass
	GTO f a	22 31 11	Flag 0 set; Pass 2 or 3
	f CLREG	31 43	Pass 1. Clear all regs.
	STO 1	33 01	and init. TW with value on Eval. Date
	g GSB fe	32 22 15	Convert Eval. Date
	CHS	42	from MM.YYYY to
	STO 5	33 05	YY. fraction & store
	RCL 1	34 01	Display value on
	h RTN	35 22	Eval. Date
	f LBL B	31 25 12	Lump Sum
030	h F? 0	35 71 00	Test for Pass
	GTO f b	22 31 12	Flag 0 set. Pass 2 or 3
	g GSB fe	32 22 15	Pass 1. Cnvt date in
	h X⇄Y	35 52	Y Reg to -n; amount in X Reg.
	f LBL 0	31 25 00	Common lump sum & series
	X	71	(-WDL)(-n) or (PMT)(-n)
	h LST X	35 82	Bring back PMT or WDL
	f X<0	31 71	PMT or WDL?
	GTO 1	22 01	WDL; Go to LBL 1
	STO + 2	33 61 02	Σ PMT
040	h R↓	35 53	(-n)(PMT) to X Reg
	STO − 4	33 51 04	Σ − (-n)(PMT)
	h RTN	35 22	Display − (n)(PMT)
	f LBL 1	31 25 01	WDL
	STO − 1	33 51 01	Σ − (-WDL)
	h R↓	35 53	(-n)(-WDL) to X Reg
	STO + 3	33 61 03	Σ (-n)(-WDL)
	h RTN	35 22	Display (n)(WDL)
	f LBL C	31 25 13	Series
	h F? 0	35 71 00	Test for Pass
050	GTO f c	22 31 13	Flag 0 set. Pass 2 or 3
	STO 6	33 06	Pass 1. Temp. store amnt.
	h R↓	35 53	N in X Reg; I in y Reg;
	h X⇄Y	35 52	date in Z Reg.
	STO X 6	33 71 06	(N)(PMT) or (N)(-WDL)
	CHS	42	-N
	h X⇄Y	35 52	I in x; -N in y Reg.

STEP	KEY ENTRY	KEY CODE	COMMENTS
	X	71	(-N)(I)
	h LST X	35 82	Bring back I
	+	61	I - NI
060	2	02	(I - NI)/24
	4	04	= (I' - NI')/2
	÷	81	
	g GSB fe	32 22 15	Calculate -n'
	+	61	$\bar{n} = -n' + [(I' - NI')/2]$
	RCL 6	34 06	Amnt in X; $\bar{n}$ in y Reg.
	GTO 0	22 00	Common lump sum & series
	g LBL fa	32 25 11	Pass 2 or 3 Initialize
	0	00	Initialize TDW & TDP
	STO 1	33 01	
070	STO 2	33 02	
	h X⇄Y	35 52	0 in y; mkt value in X
	−	51	- mkt. value in X
	g LBL f b	32 25 12	Pass 2 or 3 lump sum
	0	00	Not series
	STO 3	33 03	
	+	61	Amnt in X; date in Y
	f LBL 2	31 25 02	Common lump sum & series
	g GSB fe	32 22 15	Calc. -n
	RCL 3	34 03	(I'N - I') or 0
080	−	51	-n' - (I'N - I') or -n
	RCL (i)	34 24	$\pm PV = \pm Amount \times (1 + i/100)^{-n}$
	h X⇄Y	35 52	
	h y^x	35 63	
	X	71	
	f X<0	31 71	PMT or WDL?
	GTO 3	22 03	WDL; Go to LBL 3
	STO + 2	33 61 02	$\Sigma PV_{PMT} = TDP$
	h RTN	35 22	Display PV_{PMT}
	f LBL 3	31 25 03	WDL
090	STO − 1	33 51 01	$\Sigma - (PV_{WDL}) = TDW$
	h RTN	35 22	Display $-PV_{WDL}$
	g LBL fc	32 25 13	Pass 2 or 3 Series
	STO 4	33 04	Temp. store amount
	CLX	44	I' = I/12
	1	01	
	2	02	
	÷	81	
	X	71	I'N
	STO 3	33 03	
100	RCL (i)	34 24	$(1 + i/100)$ in X; I'N in y
	h LST X	35 82	Bring back I'
	STO − 3	33 51 03	(I'N - I') in Reg 3
	h y^x	35 63	$A = [1 + i/100]^{I'} - 1$
	1	01	
	−	51	
	h X⇄Y	35 52	A in y; I'N in X
	RCL (i)	34 24	$(1 + i/100)$ in x; I'N in y
	h X⇄Y	35 52	$(1 + i/100)$ in y; I'N in X
	h y^x	35 63	$B = (1 + i/100)^{I'N} - 1$
110	1	01	
	−	51	
	h X⇄Y	35 52	

REGISTERS

0	1 TW or TDW	2 TP or TDP	3 Temp; $1+(i_2/100)$	4 Temp.	5 Eval. Date	6 TW/TP	7 $1+(i/100)$	8 $1+(i_1/100)$	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I
					7 or 8

67 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
	÷	81	B/A
	RCL 4	34 04	Amount =
	X	71	Amount × (B/A)
	GTO 2	22 02	Common lump sum & series
	f LBL D	31 25 14	Calculate Rates
	h F? 0	35 71 00	Test for pass
	GTO f d	22 31 14	Flag 0 set. Pass 2 or 3
120	RCL 2	34 02	Pass 1. Calc. initial rate.
	RCL 1	34 01	TP/TW
	÷	81	
	STO 6	33 06	Store TP/TW
	RCL 4	34 04	$x_p = [\Sigma(PMT \times n)]/TP$
	RCL 2	34 02	
	÷	81	
	RCL 3	34 03	$x_w = [\Sigma(WDL \times n)]/TW$
	RCL 1	34 01	
	÷	81	
130	−	51	$a = x_p - x_w$
	h 1/x	35 62	$(1 + i/100) = (TP/TW)^{1/a}$
	h y^x	35 63	
	STO 7	33 07	Store $1 + (i/100)$
	h RTN	35 22	Display $1 + (i/100)$
	g LBL f d	32 25 14	Pass 2 or 3 Rates
	RCL (i)	34 24	$(1 + i/100)$ or $(1 + i_1/100)$
	RCL 6	34 06	$A = \log(TW/TP)$
	f LOG	31 53	
	RCL 6	34 06	$B = \log\left[\frac{(TW/TP)}{(TDP/TDW)}\right]$
140	RCL 2	34 02	
	÷	81	
	RCL 1	34 01	
	X	71	
	f LOG	31 53	
	÷	81	$1/b = A/B$
	h y^x	35 63	$(1 + i_1/100) = (1 + i/100)^{1/b}$
	h F? 1	35 71 01	Test for Pass 3
	GTO 5	22 05	Flag 1 set. Pass 3
	STO 8	33 08	Pass 2. Store $(1 + i_1/100)$
150	h RTN	35 22	Display $1 + (i_1/100)$
	f LBL 5	31 25 05	Pass 3. Calc. final rate
	STO 3	33 03	Store $(1 + i_2/100)$
	f −x−	31 84	Display $1 + (i_2/100)$
	RCL 8	34 08	$A = [(i_1/100) - (i/100)]^2$
	RCL 7	34 07	
	−	51	
	↑	41	
	X	71	
	h LST x	35 82	
160	RCL 3	34 03	$B = (i_1/100) - (i/100)$
	−	51	$-(i_2/100) + (i_1/100)$
	RCL 8	34 08	
	+	61	
	f X≠0	31 61	Are i, i_1, and i_2 all equal?
	GTO 6	22 06	No. Proceed
	CLX	44	Yes. Set B = 1
	1	01	
	f LBL 6	31 25 06	

STEP	KEY ENTRY	KEY CODE	COMMENTS
	÷	81	A/B
170	RCL 7	34 07	
	1	01	
	−	51	$i_c = [(i/100) + (A/B)] \times 100$
	+	61	
	EEX	43	
	2	02	
	X	71	
	h RTN	35 22	Display i_c
	f LBL E	31 25 15	Set up next pass
	h F? 0	35 71 00	Is current pass 1?
180	GTO 8	22 08	Current pass is 2 or 3
	h SF 0	35 51 00	Pass 1. Increase to 2
	h CF 1	35 61 01	Clear pass 3 flag
	7	07	Set I Reg. to 7 for i_1
	h STI	35 33	
	2	02	
	h RTN	35 22	Display pass 2
	f LBL 8	31 25 08	Current pass is 2 or 3
	h F? 1	35 71 01	Is current pass 3?
	GTO 9	22 09	Current pass is 3
190	h SF 1	35 51 01	Pass 2. Increase to 3
	8	08	Set I Reg. to 8 for i_2
	h STI	35 33	
	3	03	
	h RTN	35 22	Display Pass 3
	f LBL 9	31 25 09	Current pass is 3
	h CF 0	35 61 00	Reset to 1. Clear pass 2 or 3 flag
	h CF 1	35 61 01	Clear pass 3 flag
	1	01	
	h RTN	35 22	Display Pass 1
200			
210			
220			

LABELS

A	B	C	D	E
Start	Lump Sum	Series	Calc. Rate	Next Pass
a Pass 2 or 3 Start	b Pass 2 or 3 Lump Sum	c Pass 2 or 3 Series	d Pass 2 or 3 Calc. Rate	e Convert Date to −n
0 Pass 1 Common	1 Pass 1 WDL	2 Pass 2 or 3 Common	3 Pass 2 or 3 WDL	4
5 Pass 3 Final Rate	6 used	7	8 Current Pass 2 or 3	9 Current Pass 1

FLAGS

0	1	2	3
Pass 2 or 3	Pass 3		

SET STATUS

FLAGS	ON	OFF
0	☒	☐
1	☒	☐
2	☐	☐
3	☐	☐

TRIG		DISP	
DEG	☒	FIX	☒
GRAD	☐	SCI	☐
RAD	☐	ENG	☐
		n	4

Program Description I

Program Title DIET PLANNING

Contributor's Name Hewlett-Packard Company

Address 1000 N.E. Circle Boulevard

City Corvallis **State** Oregon **Zip Code** 97330

Program Description, Equations, Variables Most diet plans emphasize energy input, or counting calories in the food to be eaten, without enough regard for the individual's energy output, or the calories/day required. Since calculation of caloric needs is tedious, it is usually ignored, with reliance on some fixed-calorie diet instead. However, weight gain or loss is determined by the relationship between energy input and energy output.

This program estimates an individual's basal metabolism (energy requirement necessary to keep the body alive, at rest, with no weight change, at 20°C.) based upon height, weight, age, and sex. Next the required calories/day to maintain the current weight are estimated from the basal metabolism and the hours/day spent at each of five activity levels. Finally the required calories/day to lose (or gain) weight at the desired rate are estimated from the caloric content of body fat. Inputs may be in either English or metric units, and are automatically converted, if necessary. Knowledge of one's daily calorie requirement allows for more rational diet planning.

Fad diets may result in weight loss, but do not establish good eating habits which will allow maintainence of the desired weight. Good diets maintain a balance of food types, so that nutritional requirements are met. A balanced diet would contain at least 12 to 14% protein and not over 35% fat, with the rest being carbohydrate. (Note that protein and carbohydrate have 4 calories/gram, while fat has 9 calories/gram. Therefore 35% of a diet's calories may be given by about 19% by weight of fat). For best health, the carbohydrates should be primarily natural carbohydrates rather than the highly refined sugars and white flour.

The actual diet to use is left to the user. Many diet manuals have lists of the calorie content of various foods; so long as your calorie input is less than your calorie requirement, you will lose weight (and vice versa). While these calorie requirement estimates are not exact, they should help the dieter to modify diet and/or activities so as to move towards and maintain the desired weight.

Operating Limits and Warnings Individual differences may cause the estimates of basal metabolism and calorie needs to vary by ± 10% from their absolute values; however, once the deviation is established, it should remain consistent.

Age must be from 5 to 80 years, inclusive.

In this program, the term "calories" actually means "kilocalories", as is commonly the case when referring to the energy content of foods.

Program Description I

Program Title DIET PLANNING (Cont.)

Contributor's Name

Address

City **State** **Zip Code**

Program Description, Equations, Variables Programming

This program first calculates body surface area in sq. meters, using the equation:

$$SA = W^{.425} \times H^{.725} \times 0.007184$$

Weight and height values are converted to metric units before calculation, if necessary. The program then refers to tables of basal metabolism rate (BMR)/sq. meter/hour vs. age (one table for each sex). These tables (in R6) cover ages from 5 years to 80 years. Flag 1 determines whether the male or the female table is selected from R6. Since both tables start at "49" for 5 years, it was possible to store only the differences between the ages of 5, 10, 20, 40, and 80 years. These ages were chosen to match changes in the slope of the curve of BMR/m^2/hr.vs. age. The program determines in which interval the entered age lies, then performs a linear interpolation of the BMR for the exact age. Thus a table look-up with interpolation fits data that are not reducable to a simple equation.

The product of the interpolated tabular value and the surface area gives the BMR/hr., and multiplying by 24 gives the basal metabolism in calories/day.

Next, to find the total required calories/day, the number of hours at each of five levels of activity is multiplied by its estimated calorie requirement and accumulated. Total hours must equal 24, or no answer is given. Calories/hour are assumed to be at the BMR for sleeping, and at 80, 160, 240, 320 calories/hr./ 70 kg. body weight for the other four levels of activity. While very strenuous exercise may require over 320 calories/hour, such exercise is normally not sustained for one hour.

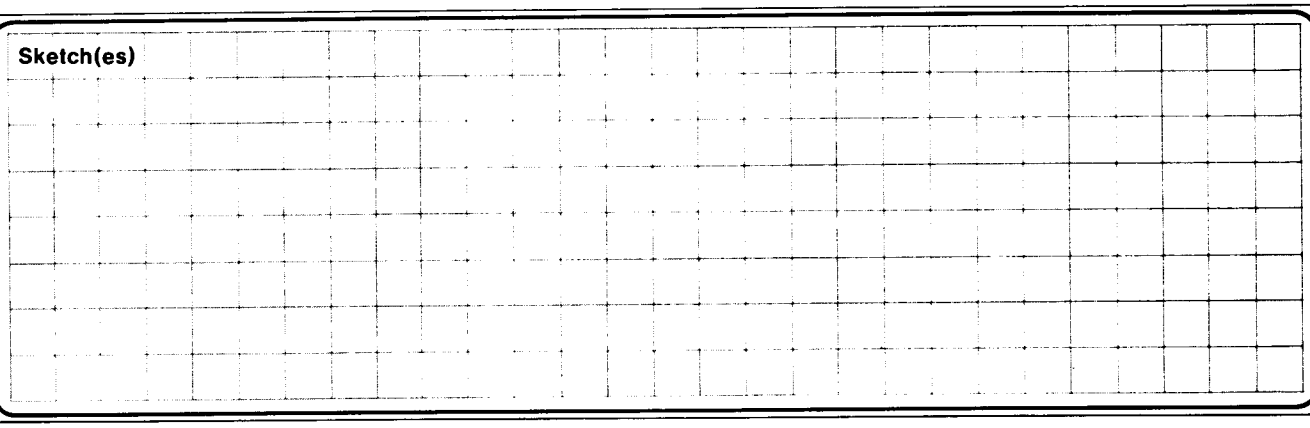
Finally, if a weight change is desired, the Δweight and the time (in weeks) to make the change are combined to calculate a new total required calories/day to give the desired rate of weight change. Body fat is assumed to contain 3,500 calories/pound (instead of the 4,100 calories/pound for pure fat) because of its water content. Kilograms are converted to pounds, if necessary.

OPERATING LIMITS AND WARNINGS

In the original reference article, note that the surface area equation has the exponents reversed in the text, but has the correct equation in line No. 305 of the BASIC language program. Also note that in the program DATA listing, the last number in line No. 359 should be "37.9" instead of "39.7".

Program Description II

Sketch(es)



Sample Problem(s) Calculate Diet Planning data in English units for a man, 70 inches height, 170 lbs. weight, 35 years old, who sleeps 8 hrs., sits 4 hrs., stands 8 hrs., walks 3 hrs., exercises 1 hr., and wants to lose 10 lbs. in 8 weeks.

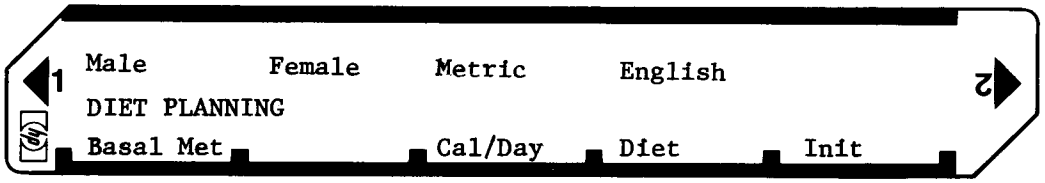
Solution:

Operation			
Enter Program, Part 1	--	--	--
Initialize	E	0.00	--
Set for Male	[f] [A]	0.00	--
Set for English units	[f] [A]	0.00	--
Ht. - 70 in., Wt. - 170 lb., Age - 35 yr.	70, [ENTER], 170, [ENTER], 35, [A]	1708.	Basal Metabolism, Cal./day
Set for Total Req. Cal./day	[C]	0.	Activity # for Hrs. to Enter
Enter 8 hrs. Sleep	8, [R/S]	1.	" " " " " "
Enter 4 hrs. Sitting	4, [R/S]	2.	" " " " " "
Enter 8 hrs. Standing	8, [R/S]	3.	" " " " " "
Enter 3 hrs. Walking	3, [R/S]	4.	" " " " " "
Enter 1 hr. Exercise	1, [R/S]	3484.	Cal./day to Maintain Weight
Diet 8 wk. for 10 lb. loss	8, [ENTER], 10, [CHS], [D]	2859.	Cal.day to Lose Weight at Rate of 10 lb. in 8 wk.

Reference(s) Martin Cl. Beattie, M.D. "BASIC Diet Planning" Interface Age Vol. 1, No. 11 October, 1976 pp. 26-42

This program is a translation of the HP-65 Users' Library Program #05351A submitted by Delmer D. Hinrichs.

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load side 1 and side 2			
2	Initialize		E	0.00
3	Set for Male or Female: Male:		f A	
	or Female		f B	
4	Set for Metric or English unit: Metric:		f C	
	or English:		f D	
5	Enter Personal Data and Calc. Basal Metabolism			
	a) Height (centimeters or inches):	Height	ENT	Height
	b) Weight (kilograms or pounds):	Weight	ENT	Weight
	c) Age (5 to 80 years):	Age	A	BM, Cal/day
6	Set for Required Cal./Day for Current Weight		C	0.
7	Enter Hours for Activity No. Displayed:			
	0) Sleeping or Resting, hours/day:	Sleep	R/S	1.
	1) Sitting, hours/day:	Sitting	R/S	2.
	2) Standing or Light Activity, hours/day:	Standing	R/S	3.
	3) Walking or Moderate Activity, hrs./day:	Walking	R/S	4.
	4) Exercising or Heavy Work, hours/day:	Exercise	R/S	Req.Cal/day
	(If total hours do not equal 24, program			
	will return to Step 6 and display "0.".			
	If this occurs, reenter activity hours			
	correctly.)			
8	Enter Diet Data, to Calculate Required			
	Cal./day for desired rate of weight loss			
	(or gain):			
	a) Length of Diet Plan, weeks:	Length	ENT	Length
	b) Desired Weight Change (kilo. or pound):	ΔWeight	D	Reg.Cal/day
	(ΔWeight must be negative for loss,			
	positive for gain.)			
9	For a New Problem, go to Step 3.			
	Notes:			
	1. As required calories/day will change with weight			
	change, results should be recalculated each week			
	for new weight.			
	2. Steps (5), (6&7), and (8) may be recalculated			
	as desired to see the effect of changing parameters.			

97 Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBL E	21 15	Initialize:
002	.	-62	
003	7	07	
004	2	02	
005	5	05	
006	STO 1	35 01	Height Exponent
007	.	-62	
008	4	04	
009	2	02	
010	5	05	
011	STO 2	35 02	Weight Exponent
012	7	07	
013	1	01	
014	8	08	
015	4	04	
016	EEX	-23	
017	6	06	
018	CHS	-22	
019	STO 3	35 03	Surface Area Const
020	2	02	
021	.	-62	
022	5	05	
023	4	04	
024	STO 4	35 04	Centimeters/Inch
025	2	02	
026	.	-62	
027	2	02	
028	STO 5	35 05	Pounds/Kilogram
029	.	-62	
030	0	00	
031	5	05	
032	6	06	
033	2	02	
034	3	03	
035	0	00	
036	7	07	
037	7	07	
038	0	00	
039	4	04	
040	CHS	-22	Condensed BMR/m^2.
041	STO 6	35 06	Table for Mal&Fem
042	*LBL C	21 13	Calculate Req.
043	RCL 7	36 07	Cal./Day for
044	0	00	Current Weight
045	R/S	51	Stop & Display "0."
046	STO 9	35 09	Hours of Sleep
047	X≠Y	-41	
048	R↓	-31	
049	×	-35	
050	1	01	Increment Hrs. &
051	GSB B	23 12	Cal. for each
052	2	02	Activity
053	GSB B	23 12	
054	3	03	
055	GSB B	23 12	
056	4	04	
057	GSB B	23 12	
058	2	02	Total Activity Hrs.
059	4	04	Total Hrs. = 24?
060	RCL 9	36 09	
061	X≠Y?	16-32	
062	GTO C	22 13	No, Ret. and Try Aga.
063	R↓	-31	Yes, Recall Cal/Day
064	R↓	-31	
065	STO 9	35 09	
066	PRTX	-14	
067	RTN	24	Display Req. Cal/Day
068	*LBL D	21 14	Calc. Req Cal/Day
069	F0?	16 23 00	Metric?
070	GSB 4	23 04	Yes, Convert to lb.
071	X≠Y	-41	No, Continue
072	÷	-24	Δ Pounds/Week
073	5	05	
074	EEX	-23	
075	2	02	
076	×	-35	Δ Calories/Day
077	RCL 9	36 09	Cal/Day for Cur.Wt.
078	+	-55	Display Diet Cal/Day
079	PRTX	-14	
080	RTN	24	
081	*LBL 4	21 04	
082	RCL 5	36 05	
083	×	-35	
084	RTN	24	
085	*LBL B	21 12	Subroutine to Incre.
086	R/S	51	hours & Calories
087	ST+ 9	35-55 09	Display Activity No.
088	X≠Y	-41	Increment Hours
089	8	08	
090	0	00	
091	×	-35	Activity No. X 80
092	RCL 6	36 06	Weight Factor
093	×	-35	
094	×	-35	
095	+	-55	Increment Calories
096	RTN	24	
097	*LBL A	21 11	Calc. Basal Metab:
098	STO 9	35 09	Store Age
099	R↓	-31	
100	F0?	16 23 00	Metric?
101	GTO 1	22 01	Yes, Continue
102	RCL 5	36 05	No, Convert to
103	÷	-24	Kilograms
104	*LBL 1	21 01	
105	STO 8	35 08	
106	RCL 2	36 02	Weight (.425)
107	Y^	31	
108	X≠Y	-41	
109	F0?	16 23 00	Metric?
110	GTO 2	22 02	Yes, Continue
111	RCL 4	36 04	No, Convert to
112	×	-35	Centimeters

REGISTERS

0	1 Ht. Exp.	2 Wt. Exp.	3 SA Factor	4 CM/Inch	5 Lb./Kg.	6 BMR/m^2 Table	7 BMR Cal./Hr.	8 Wt. Fac.	9 Used
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I

97 Program Listing II

STEP	KEY ENTRY	KEY CODE	COMMENTS
113	*LBL2	21 02	
114	RCL1	36 01	
115	Y^x	31	Height (.425)
116	RCL3	36 03	
117	×	-35	
118	×	-35	
119	STO7	35 07	Surface Area, Sq. m
120	1	01	
121	RCL6	36 06	BMR/m^2 Tables, M&F
122	F1?	16 23 01	
123	GTO3	22 03	Male?
124	EEX	-23	
125	5	05	No, Enter 10^5
126	*LBL3	21 03	
127	×	-35	Multiply by 1 or 10^5
128	FRC	16 44	
129	4	04	
130	9	09	49, First Tabular BMR/m^2
131	ENT↑	-21	
132	5	05	
133	ENT↑	-21	5, First Tabular Age
134	*LBL0	21 00	
135	CLX	-51	Loop to Find Portio of Table to Use
136	GSB5	23 05	
137	R↑	16-31	
138	LSTX	16-62	
139	FRC	16 44	
140	R↓	-31	
141	+	-55	
142	X≷Y	-41	
143	RCL9	36 09	
144	X>Y?	16-34	Corect Part of Tab.
145	GTO0	22 00	No, Ret. & Try Agn.
146	X≷Y	-41	
147	-	-45	
148	LSTX	16-63	
149	÷	-24	Interpolate
150	GSB6	23 06	
151	×	-35	
152	LSTX	16-63	
153	+	-55	
154	+	-55	
155	RCL7	36 07	Surface Area
156	×	-35	
157	STO7	35 07	BMR, Calories/Hour
158	7	07	
159	0	00	
160	ST÷8	35-24 08	Weight Fac. (Kg./70)
161	CLX	-51	
162	2	02	
163	4	04	
164	×	-35	BM, Calories/Day
165	DSP0	-63 00	
166	PRTX	-14	
167	RTN	24	
168	*LBL5	21 05	
169	R↓	-31	Display Basal Metab
170	*LBL6	21 06	Subrountine
171	2	02	
172	×	-35	
173	EEX	-23	
174	1	01	
175	R↑	16-31	
176	×	-35	
177	INT	16 34	
178	RTN	24	
179	*LBLa	21 16 11	Male
180	SF1	16 21 01	
181	RTN	24	
182	*LBLb	21 16 12	Female
183	CF1	16 22 01	
184	RTN	24	
185	*LBLc	21 16 13	Metric
186	SF0	16 21 00	
187	RTN	24	
188	*LBLd	21 16 14	English
189	CF0	16 22 00	
190	RTN	24	
191	R/S	51	

200

210

220

LABELS

A Basal	B	C Cal/Day	D Diet	E Init.	0 Units
a Male	b Female	c Metric	d English	e	1 Sex
0 Used	1 Used	2 Used	3	4 Used	2
5 Used	6 Used	7	8	9	3

FLAGS

SET STATUS

FLAGS			TRIG		DISP	
	ON	OFF				
0	☐	☐	DEG	☐	FIX	☐
1	☐	☐	GRAD	☐	SCI	☐
2	☐	☐	RAD	☐	ENG	☐
3	☐	☐			n_____	

Hewlett-Packard Software

In terms of power and flexibility, the problem-solving potential of the Hewlett-Packard line of fully programmable calculators is nearly limitless. And in order to see the practical side of this potential, we have several different types of software to help save you time and programming effort. Every one of our software solutions has been carefully selected to effectively increase your problem-solving potential. Chances are, we already have the solutions you're looking for.

Application Pacs

To increase the versatility of your fully programmable Hewlett-Packard calculator, HP has an extensive library of "Application Pacs". These programs transform your HP-67 and HP-97 into specialized calculators in seconds. Each program in a pac is fully documented with commented program listing, allowing the adoption of programming techniques useful to each application area. The pacs contain 20 or more programs in the form of prerecorded cards, a detailed manual, and a program card holder. Every Application Pac has been designed to extend the capabilities of our fully programmable models to increase your problem-solving potential.

You can choose from:

Statistics
Mathematics
Electrical Engineering
Business Decisions
Clinical Lab and Nuclear Medicine
Mechanical Engineering
Surveying
Civil Engineering
Navigation
Games

Users' Library

The main objective of our Users' Library is dedicated to making selected program solutions contributed by our HP-67 and HP-97 users available to you. By subscribing to our Users' Library, you'll have at your fingertips, literally hundreds of different programs. No longer will you have to: research the application; program the solution; debug the program; or complete the documentation. Simply key your program to obtain your solution. In addition, programs from the library may be used as a source of programming techniques in your application area.

A one-year subscription to the Library costs $9.00. You receive: a catalog of contributed programs; catalog updates; and coupons for three programs of your choice (a $9.00 value).

Users' Library Solutions Books

Hewlett-Packard recently added a unique problem-solving contribution to its existing software line. The new series of software solutions are a collection of programs provided by our programmable calculator users. Hewlett-Packard has currently accepted over 6,000 programs for our Users' Libraries. The best of these programs have been compiled into 40 Library Solutions Books covering 39 application areas (including two game books).

Each of the Books, containing up to 15 programs without cards, is priced at $10.00, a savings of up to $35.00 over single copy cost.

The Users' Library Solutions Books will compliment our other applications of software and provide you with a valuable new tool for program solutions.

Options/Technical Stock Analysis
Portfolio Management/Bonds & Notes
Real Estate Investment
Taxes
Home Construction Estimating
Marketing/Sales
Home Management
Small Business
Antennas
Butterworth and Chebyshev Filters
Thermal and Transport Sciences
EE (Lab)
Industrial Engineering
Aeronautical Engineering
Control Systems
Beams and Columns
High-Level Math
Test Statistics
Geometry
Reliability/QA
Medical Practitioner
Anesthesia
Cardiac
Pulmonary
Chemistry
Optics
Physics
Earth Sciences
Energy Conservation
Space Science
Biology
Games
Games of Chance
Aircraft Operation
Avigation
Calendars
Photo Dark Room
COGO-Surveying
Astrology
Forestry

HOME MANAGEMENT

This book contains something for everyone. For example, it includes a program to help select the optimum tax schedule; another which evaluates the most economical automobile to own; one which displays telephone time and charges like a taxicab meter; and even a program to aid you in planning a diet.

INCOME TAX PLANNING — I

TRUE COST OF INSURANCE POLICY

AUTOMOBILE COST/TIRE COST COMPARISON

COMPARISON SHOPPING

TIME & CHARGES RUNNING TOTAL

RECONCILE CHECKING ACCOUNT

SAVINGS ACCOUNT COMPOUNDED DAILY

ACCUMULATED INTEREST/REMAINING BALANCE

STOCK PORTFOLIO VALUATION AND DATA CARD

TRUE ANNUAL GROWTH RATE OF AN INVESTMENT PORTFOLIO

DIET PLANNING

HEWLETT hp PACKARD

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

Reorder No. 00097-14031 Printed in U.S.A. 00097-90206

Revision E 3-79