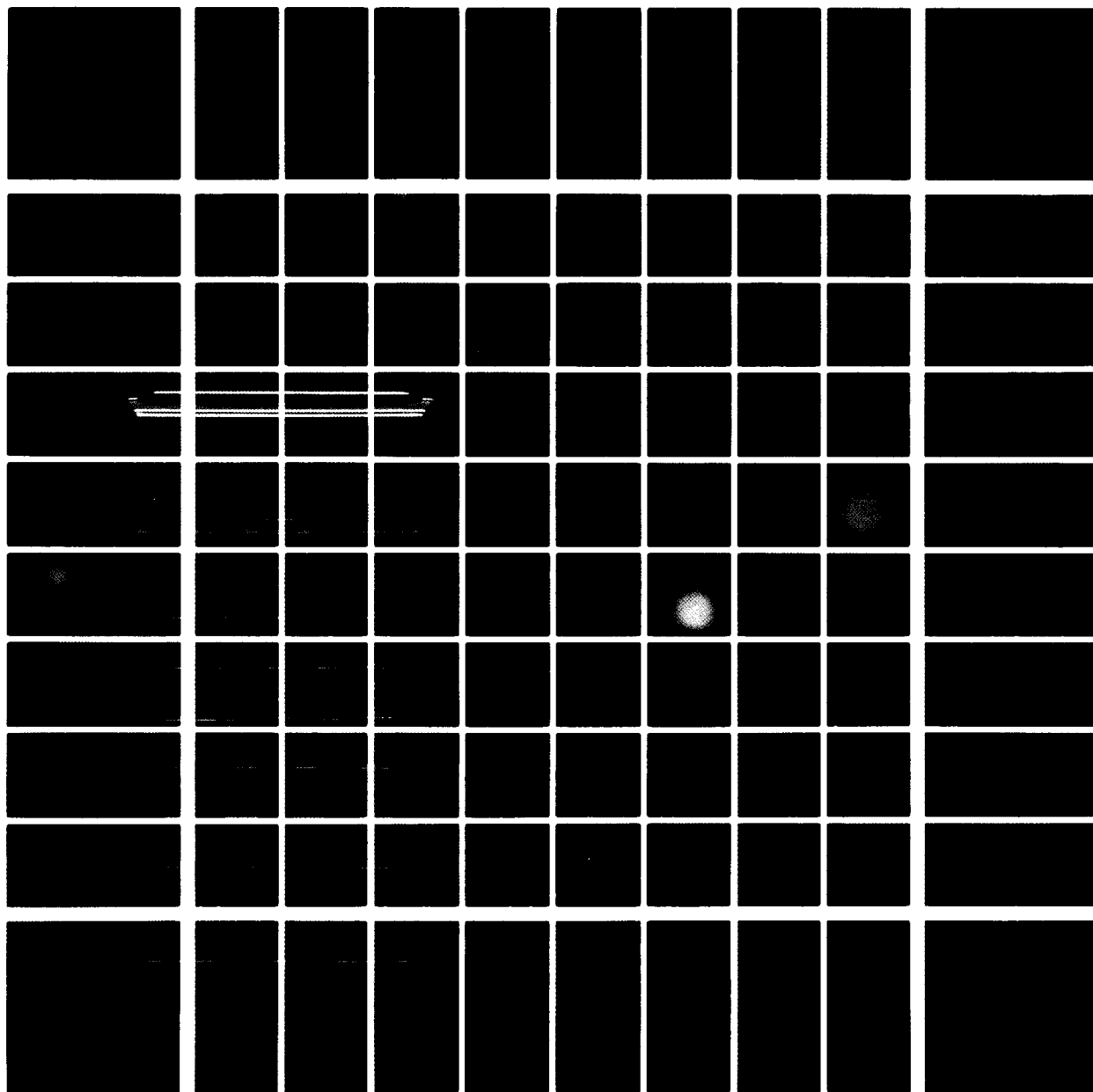


HEWLETT-PACKARD

HP-41

USERS' LIBRARY SOLUTIONS  
High Level Math

Includes barcode for easy software entry.



## NOTICE

The program material contained herein is supplied without representation or warranty of any kind. Hewlett-Packard Company therefore assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from the use of this program material or any part thereof.

## INTRODUCTION

This HP-41C Solutions book was written to help you get the most from your calculator. The programs were chosen to provide useful calculations for many of the common problems encountered.

They will provide you with immediate capabilities in your everyday calculations and you will find them useful as guides to programming techniques for writing your own customized software. The comments on each program listing describe the approach used to reach the solution and help you follow the programmer's logic as you become and expert on your HP calculator.

### KEYING A PROGRAM INTO THE HP-41C

There are several things that you should keep in mind while you are keying in programs from the program listings provided in this book. The output from the HP 82143A printer provides a convenient way of listing and an easily understood method of keying in programs without showing every keystroke. This type of output is what appears in this handbook. Once you understand the procedure for keying programs in from the printed listings, you will find this method simple and fast. Here is the procedure:

1. At the end of each program listing is a listing of status information required to properly execute that program. Included is the SIZE allocation required. Before you begin keying in the program, press **XEQ** **ALPHA** SIZE **ALPHA** and specify the allocation (three digits; e.g., 10 should be specified as 010).

Also included in the status information is the display format and status of flags important to the program. To ensure proper execution, check to see that the display status of the HP-41C is set as specified and check to see that all applicable flags are set or clear as specified.

2. Set the HP-41C to PRGM mode (press the **PRGM** key) and press **▀** **GTO** **◻** **◻** to prepare the calculator for the new program.

3. Begin keying in the program. Following is a list of hints that will help you when you key in your programs from the program listings in this handbook.

- a. When you see " (quote marks) around a character or group of characters in the program listing, those characters are ALPHA. To key them in, simply press **ALPHA** , key in the characters, then press **ALPHA** again. So "SAMPLE" would be keyed in as **ALPHA** "SAMPLE" **ALPHA** .
- b. The diamond in front of each LBL instruction is only a visual aid to help you locate labels in the program listings. When you key in a program, ignore the diamond.
- c. The printer indication of divide sign is /. When you see / in the program listing, press **+** .
- d. The printer indication of the multiply sign is  $\times$  . When you see  $\times$  in the program listing, press **x** .
- e. The † character in the program listing is an indication of the **APPEND** function. When you see †, press **▀** **APPEND** in ALPHA mode (press **▀** and the K key).
- f. All operations requiring register addresses accept those addresses in these forms:

nn (a two-digit number)

IND nn (INDIRECT: **▀** , followed by a two-digit number)

X, Y, Z, T, or L (a STACK address: **◻** followed by X, Y, Z, T, or L)

IND X, Y, Z, T or L (INDIRECT stack: **▀** **◻** followed by X, Y, Z, T, or L)

Indirect addresses are specified by pressing **▀** and then the indirect address. Stack addresses are specified by pressing **◻** followed by X, Y, Z, T, or L. Indirect stack addresses are specified by pressing **▀** **◻** and X, Y, Z, T, or L.

#### Printer Listing

```

01♦LBL "SAM
PLE"
02 "THIS IS
A "
03 †SAMPLE
"
04 AVIEW
05 6
06 ENTER†
07 -2
08 /
09 ABS
10 STO IND
L
11 "R3="
12 ARCL 03
13 AVIEW
14 RTN
    
```

#### Keystrokes

```

▀ LBL ALPHA SAMPLE ALPHA
ALPHA THIS IS A ALPHA
ALPHA ▀ APPEND SAMPLE
▀ AVIEW ALPHA
6
ENTER+
2 CHS
+
XEQ ALPHA ABS ALPHA
STO ▀ ◻ L
ALPHA R3= ▀ ARCL 03
▀ AVIEW
ALPHA
▀ RTN
    
```

#### Display

```

01 LBLT SAMPLE
02T THIS IS A
03T †SAMPLE
04 AVIEW
05 6
06 ENTER†
07 -2
08 /
09 ABS
10 STO IND L
11T R3=
12 ARCL 03
13 AVIEW
14 RTN
    
```

## TABLE OF CONTENTS

1.	SINE, COSINE, EXPONENTIAL INTEGRALS . . . . .	1
	Calculate $\int_0^x \frac{\sin t}{t} dt$ , $\gamma + \ln x + \int_0^x \frac{\cos t - 1}{t} dt$ , and $\int_x^\infty \frac{e^{-t}}{t} dt$ .	
*2.	EIGENVALUES/VECTORS OF 3RD - ORDER SYSTEMS . . . . .	7
	For a given 3rd order matrix with distinct real eigenvalues the program calculates the eigenvalues and eigenvectors. The first (largest) eigenvalue and eigenvector are calculated by the power method, while the 2nd and 3rd are calculated by the deflation method.	
*3.	EIGENVALUES FOR 3RD ORDER SYSTEMS . . . . .	18
	Program calculates the eigenvalues of a 3rd order system described by $Az - \lambda x$ .	
4.	CHEBYSHEV, LEGENDRE, HERMITE, AND LAGUERRE POLYNOMIALS . . . . .	27
	Calculates the polynomials by using recurrence equations.	
5.	SIXTEEN-POINT GAUSSIAN QUADRATURE . . . . .	35
	Computes approximations for integrals over finite or infinite intervals.	
6.	GAMMA FUNCTION . . . . .	42
	Program approximates the value of the gamma function $\Gamma(x)$ for $1 < x < 70$ .	
7.	BESSEL FUNCTIONS, ERROR FUNCTION . . . . .	48
	Computes the Bessel functions, $J_n(x)$ and $I_n(x)$ and the error function.	
*8.	CHARACTERISTIC EQUATION OF 4 X 4 MATRIX . . . . .	58
	Calculates the coefficients to the characteristic equation. $\lambda^4 + r_1\lambda^3 + r_2\lambda^2 + r_3\lambda + r_4 = 0$	
**9.	4 X 4 MATRIX OPERATIONS . . . . .	67
	Computes determinant and inverse of a 4x4 matrix, solves four simultaneous equations and four unknowns, by Gaussian elimination.	

\* This program requires 1 extra memory module  
 \*\* This program requires 2 extra memory modules

## SINE, COSINE, EXPONENTIAL INTEGRALS

This program will calculate the following integrals:

### Sine Integral

$$Si(x) = \sum_{a=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1) \cdot (2n+1)!}$$

$$= \int_0^x \frac{\sin t}{t} dt$$

$$Si(-x) = Si(x)$$

### Cosine Integral

$$Ci(x) = \gamma + \ln x + \int_0^x \frac{\cos t - 1}{t} dt$$

$$= \gamma + \ln x + \sum_{n=1}^{\infty} \frac{(-1)^n x^{2n}}{2n(2n)!}$$

$$Ci(-x) = Ci(x) - i\pi \text{ for } x > 0$$

NOTE: For  $Si(x)$  and  $Ci(x)$  the accuracy of the answer decreases as  $x$  increases. For  $x=10$ , answer is accurate to the seventh decimal place. For  $x$  around 20, answers are accurate to about the second decimal place.

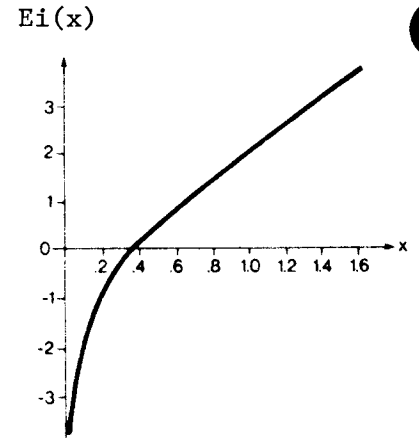
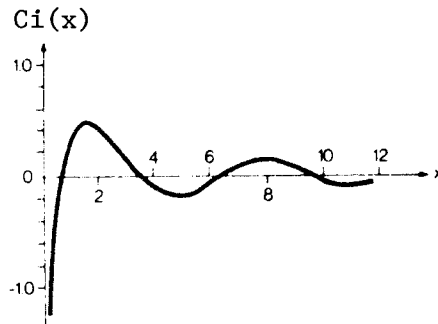
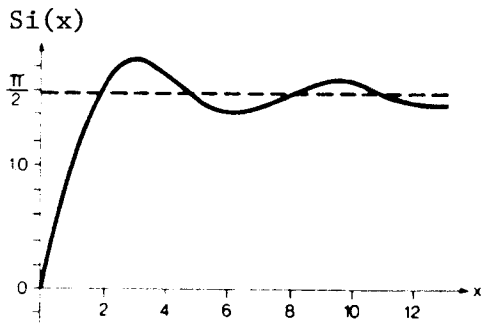
### Exponential Integral

$$Ei(x) = \int_{-\infty}^x \frac{e^t}{t} dt = \gamma + \ln x + \sum_{n=1}^{\infty} \frac{x^n}{n n!}$$

where  $x > 0$ , and  $\gamma = 0.5772156649$  is Euler's constant.

NOTE: For  $Ei(x)$  when  $x$  is too large, computing a new term of the series might cause an overflow. In that case, display shows "OUT OF RANGE" and the program halts.

The program computes successive partial sums of the series. When two consecutive partial sums are equal, the value is used as the sum of the series.



Reference: Abramowitz, *Handbook of Mathematical Functions*, National Bureau of Standards, 1968.

Examples:

1. Find Si (0.69)
2. Find Si (9.8)
3. Find Ci (1.38)
4. Find Ci (5)
5. Find Ei (1.59)
6. Find Ei (0.61)

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 004

[XEQ] [ALPHA] SI [ALPHA]

1. .69 [A]
2. 9.8 [A]
3. 1.38 [B]
4. 5 [B]
5. 1.59 [C]
6. .61 [C]

Display:

SI(0.69)=0.67

SI(9.80)=1.67

CI(1.38)=0.46

CI(5.00)=-0.19

EI(1.59)=3.57

EI(0.61)=0.80







# REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS			STATUS				
00	$-x^2, x$	50	SIZE	004	TOT. REG.	22	USER MODE
01	temp. storage		ENG		FIX	2	SCI
02	temp. storage		DEG		RAD		GRAD
03	x						ON <input checked="" type="checkbox"/> OFF <input type="checkbox"/>
05		55	FLAGS				
			#	INIT S/C	SET INDICATES	CLEAR INDICATES	
			27		User On	User Off	
10		60					
15		65					
20		70					
25		75					
30		80					
35		85					
40		90	ASSIGNMENTS				
			FUNCTION	KEY	FUNCTION	KEY	
			$S_i(x)$	A			
			$C_i(x)$	B			
			$E_i(x)$	C			
45		95					

SINE COSINE  
EXPONENTIAL INTEGRALS  
PROGRAM REGISTERS NEEDED: 18

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 1 (1 : 6)



ROW 2 (7 : 18)



ROW 3 (19 : 27)



ROW 4 (28 : 39)



ROW 5 (40 : 49)



ROW 6 (49 : 52)



ROW 7 (52 : 63)



ROW 8 (63 : 70)



ROW 9 (71 : 75)



ROW 10 (76 : 78)



**EIGENVALUES/VECTORS OF 3RD-ORDER  
SYSTEMS W/DISTINCT REAL EIGENVALUES**  
(This program requires 1 memory module)

For a system matrix A, the eigenvalues are found from  $Ax = \lambda x$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad v_n = \begin{bmatrix} v_{n,1} \\ v_{n,2} \\ v_{n,3} \end{bmatrix}$$

(a) Power method

Assume the eigenvalues of A are  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  where  $|\lambda_1| > |\lambda_2| \geq |\lambda_3|$ .  
Now let A operate repeatedly on a vector v, which we express as a  
linear combination of the eigenvectors  $v = c_1 v_1 + c_2 v_2 + c_3 v_3$  then

$$Av = c_1 Av_1 + c_2 Av_2 + c_3 Av_3 = \lambda_1 (c_1 v_1 + c_2 \frac{\lambda_2}{\lambda_1} v_2 + c_3 \frac{\lambda_3}{\lambda_1} v_3)$$

$$A^p v = \lambda_1^p [c_1 v_1 + c_2 (\frac{\lambda_2}{\lambda_1})^p v_2 + c_3 (\frac{\lambda_3}{\lambda_1})^p v_3]$$

$$\text{therefore } \lambda_1 = \lim_{p \rightarrow \infty} \frac{(A^{p+1} v_1)}{(A^p v_1)}$$

(b) For deflation method, refer to reference (2).

NOTE: Program only works for systems with distinct real eigenvalues and a "good" guess of the initial eigenvector  $v_1$ . If a first component of the eigenvectors is zero, then it is necessary to do similarity transformations in order to use this program.

Reference: Charles Cullen, *Matrices and Linear Transformations*.  
Addison-Wesley Pub. Company, March 1967

Carl-Erik Froberg, *Intro. to Numerical Analysis*.  
Addison-Wesley Pub. Company, 1969

Examples:

$$\text{Given } A = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix} \text{ and } v_1 \text{ guess} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \text{ find } \lambda_1, v_1, \lambda_2, v_2, \lambda_3, \text{ and } v_3.$$

## Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 028  
 [///] [FIX] 4  
 [XEQ] [ALPHA] EVV [ALPHA]  
 3 [CHS] [R/S]  
 2 [R/S]  
 0 [R/S]  
 1 [R/S]  
 3 [CHS] [R/S]  
 1 [R/S]  
 0 [R/S]  
 2 [R/S]  
 3 [CHS] [R/S]  
 [C]  
 1 [R/S]  
 1 [R/S]  
 1 [R/S]  
 [R/S]  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*

## Display:

a1,1=?  
 a2,1=?  
 a3,1=?  
 a1,2=?  
 a2,2=?  
 a3,2=?  
 a1,3=?  
 a2,3=?  
 a3,3=?  
 READY  
 V1,1=?  
 V1,2=?  
 V1,3=?  
 READY  
 LAM.1=-5.0000  
 V1,1=1.0000  
 V1,2=-2.0000  
 V1,3=1.0000  
 LAM.2=-3.0000  
 V2,1=1.0000  
 V2,2=-1.0000E-9  
 V2,3=-1.0000  
 LAM.3=-1.0000  
 V3,1=1.0000  
 V3,2=2.0000  
 V3,3=1.0000

\* [R/S] is omitted when the printer is present.

# User Instructions

				SIZE: 028
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load the program.			
2	Input the matrix A.		[XEQ] EVV	a1,1=?
		a <sub>1,1</sub>	[R/S]	a2,1=?
		a <sub>2,1</sub>	[R/S]	a3,1=?
		⋮	⋮	⋮
		a <sub>2,3</sub>	[R/S]	a3,3=?
		a <sub>3,3</sub>	[R/S]	READY
3	View and/or correct matrix A. While		[B]	a1,1=( )
	viewing any element, if a number is		[R/S]	a2,1=( )
	entered before pressing [R/S], it will		⋮	⋮
	replace the currently displayed element.		[R/S]	a3,3=( )
	This mode may be exited at any point.		[R/S]	READY
4	Print matrix A. If the printer is		[///] [b]	a1,1=( )
	not attached, this function will act			⋮
	just as [B].			READY
5	Input v <sub>1</sub> . This step must be performed		[C]	v1,1=?
	regardless of whether or not v <sub>1</sub> is	v <sub>1,1</sub>	[R/S]	v1,2=?
	different from that of the previous	v <sub>1,2</sub>	[R/S]	v1,3=?
	calculation.	v <sub>1,3</sub>	[R/S]	READY
6	Print v <sub>1</sub> .		[///] [C]	v1,1=( )
				⋮
				READY
7	Calculate v <sub>1</sub> , v <sub>2</sub> , v <sub>3</sub> , λ <sub>1</sub> , λ <sub>2</sub> , λ <sub>3</sub> . This		[R/S]	LAM.1=( )
	step may only be performed if "READY" is		[R/S]*	v1,1=( )
	in the display.		[R/S]*	v1,2=( )
			[R/S]*	v1,3=( )
			[R/S]*	LAM.2=( )



# Program Listings

01♦LBL "EVV"		51 "F?"	
02 SF 21		52 FIX 4	
03 SF 27		53 SF 29	
04♦LBL A		54 RCL IND	
05 CF 05	Input A matrix	00	
06 CF 06		55 FS? 05	
07 CF 07		56 ARCL X	
08 GTO 01		57 FC? 07	
09♦LBL b		58 PROMPT	
10 ADV		59 FS? 07	
11 SF 05	Print A matrix	60 AVIEW	
12 CF 06		61 STO IND	
13 SF 07		00	
14 GTO 01		62 ISG 00	
15♦LBL C		63 CLD	
16 CF 05	Input v <sub>1</sub>	64 ISG 11	
17 SF 06		65 GTO 00	
18 CF 07		66 FS? 07	
19 GTO 01		67 ADV	
20♦LBL c		68 FS? 06	
21 ADV	Print v <sub>1</sub>	69 GTO 02	
22 SF 05		70 1.003	
23 SF 06		71 STO 11	
24 SF 07		72 ISG 10	
25 GTO 01		73 GTO 00	
26♦LBL B		74♦LBL 02	
27 SF 05	View A matrix	75 "READY"	"READY" prompt
28 CF 06		76 PROMPT	
29 CF 07		77 "A"	
30♦LBL 01		78 ASTO 26	
31 1.003		79♦LBL 03	
32 STO 10	Initialize	80 RCL 01	
33 STO 11	Input/View	81 RCL 04	
34 1	routine	82 RCL 07	
35 FS? 06		83 XEQ 10	
36 20		84 STO 23	
37 STO 00		85 RCL 02	
38♦LBL 00		86 RCL 05	
39 FIX 0		87 RCL 08	
40 CF 29		88 XEQ 10	
41 "a"	Common Input/	89 STO 24	
42 FS? 06	View routine	90 RCL 03	
43 "V1,"		91 RCL 06	
44 ARCL 11		92 RCL 09	
45 FC? 06		93 XEQ 10	
46 "F,"		94 STO 25	
47 FC? 06		95 RCL 23	
48 ARCL 10		96 RCL 20	
49 "F="		97 /	
50 FC? 05		98 RCL 26	
		99 X=Y?	
			Iterative routine to find $\lambda_1$

# Program Listings

100 GTO 04		151 RCL 08	
101 X<>Y		152 RCL 21	
102 STO 26		153 RCL 07	
103 RCL 23		154 *	
104 STO 20		155 -	
105 RCL 24		156 STO 18	
106 STO 21		157 RCL 09	
107 RCL 25		158 RCL 22	
108 STO 22		159 RCL 07	
109 GTO 03		160 *	
110 LBL 10		161 -	
111 RCL 22		162 STO 19	
112 *	Interation	163 RCL 16	
113 RCL 21	subroutine	164 +	
114 ST* Z		165 STO 25	
115 RDN		166 X↑2	
116 RCL 20		167 RCL 16	
117 ST* T		168 RCL 19	
118 RDN		169 *	
119 +		170 RCL 18	
120 +		171 RCL 17	
121 RTN		172 *	
122 LBL 04		173 -	
123 CF 09	Calculate v <sub>1</sub>	174 4	
124 1		175 *	
125 STO 00		176 -	
126 STO 27		177 SQRT	
127 STO 20		178 2	
128 RCL 24		179 /	
129 RCL 23		180 STO 23	
130 /		181 RCL 25	
131 STO 21		182 2	
132 RCL 25		183 /	
133 RCL 23		184 STO 24	
134 /		185 RCL 23	
135 STO 22		186 -	
136 RCL 26		187 STO 10	
137 RDN		188 RCL 24	
138 XEQ 06		189 RCL 23	
139 RCL 05		190 +	
140 RCL 21	Calculate λ <sub>2</sub>	191 STO 13	
141 RCL 04		192 LBL 08	
142 *		193 RCL 10	
143 -		194 RCL 16	
144 STO 16		195 -	
145 RCL 06		196 STO 12	
146 RCL 22		197 RCL 18	
147 RCL 04		198 STO 11	
148 *		199 X<>Y	
149 -		200 RCL 04	
150 STO 17		201 RCL 07	
			Common routine to calculate v <sub>2</sub> and v <sub>3</sub>



# Program Listings

```

202 RDN
203 X<>Y
204 R↑
205 *
206 RDN
207 *
208 R↑
209 +
210 RCL 10
211 RCL 26
212 -
213 /
214 ST/ 11
215 ST/ 12
216 RCL 21
217 ST+ 11
218 RCL 22
219 ST+ 12
220 RCL 10
221 RCL 20
222 RCL 11
223 RCL 12
224 FS? 09
225 GTO 06
226 XEQ 06
227 SF 09
228 RCL 10
229 X<> 13
230 STO 10
231 RCL 11
232 RCL 12
233 RCL 14
234 RCL 15
235 STO 12
236 RDN
237 STO 11
238 RDN
239 STO 15
240 RDN
241 STO 14
242 GTO 08
243♦LBL 06
244 CF 08
245 "LAM."
246 XEQ 07
247 ARCL T
248 AVIEW
249 SF 08
250 "V"
251 XEQ 07
252 ARCL Z
    
```

```

253 AVIEW
254 "V"
255 XEQ 07
256 ARCL Y
257 AVIEW
258 "V"
259 XEQ 07
260 ARCL X
261 AVIEW
262 ISG 00
263 CLD
264 1
265 STO 27
266 ADV
267 RTN
268♦LBL 07
269 FIX 0
270 CF 29
271 ARCL 00
272 FC? 08
273 GTO 07
274 "F,"
275 ARCL 27
276 ISG 27
277♦LBL 07
278 FIX 4
279 SF 29
280 "F="
281 .END.
    
```

Output  
subroutine

Output routine

80

90

00



EIGENVALUES/VECTORS OF  
3RD-ORDER SYSTEMS  
PROGRAM REGISTERS NEEDED: 69

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 1 (1 : 4)



ROW 2 (5 : 11)



ROW 3 (12 : 18)



ROW 4 (18 : 25)



ROW 5 (25 : 31)



ROW 6 (31 : 39)



ROW 7 (40 : 45)



ROW 8 (45 : 50)



ROW 9 (51 : 56)



ROW 10 (57 : 64)



ROW 11 (65 : 70)



ROW 12 (70 : 75)



ROW 13 (76 : 84)



ROW 14 (84 : 93)



ROW 15 (93 : 100)



ROW 16 (100 : 107)



ROW 17 (107 : 114)



ROW 18 (115 : 124)



EIGENVALUES/VECTORS OF  
3RD-ORDER SYSTEMS

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 19 (125 : 132)



ROW 20 (132 : 139)



ROW 21 (140 : 149)



ROW 22 (150 : 158)



ROW 23 (159 : 167)



ROW 24 (168 : 177)



ROW 25 (178 : 186)



ROW 26 (187 : 196)



ROW 27 (197 : 208)



ROW 28 (209 : 217)



ROW 29 (217 : 225)



ROW 30 (225 : 233)



ROW 31 (234 : 244)



ROW 32 (245 : 249)



ROW 33 (250 : 255)



ROW 34 (256 : 262)



ROW 35 (263 : 271)



ROW 36 (272 : 278)



EIGENVALUES/VECTORS OF  
3RD-ORDER SYSTEMS

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 37 (278 : 281)



## EIGENVALUES FOR 3RD ORDER SYSTEM

(Requires 1 memory module)

This program determines the eigenvalues of a 3rd order system described by  $Ax = \lambda x$ , i.e.,

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Eigenvalues  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  are solved from

$$\det (\lambda I - A) = 0$$

Roots for the cubic equation are solved by using the exact formula.

Example: Find the eigenvalues for:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & -0.5 \end{bmatrix}$$

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 022

[XEQ] [ALPHA] EV [ALPHA]

1 [R/S]

0 [R/S]

0 [R/S]

0 [R/S]

0 [R/S]

1 [R/S]

0 [R/S]

1 [CHS] [R/S]

.5 [CHS] [R/S]

[C]

[R/S]\*

[R/S]\*

Display:

a1,1=?

a2,1=?

a3,1=?

a1,2=?

a2,2=?

a3,2=?

a1,3=?

a2,3=?

a3,3=?

READY

RLAM.1=-0.2500

ILAM.1=-0.9682

RLAM.2=-0.2500

Keystrokes:

[R/S]\*

[R/S]\*

Display:

ILAM.2=0.9682

LAM.3=1.0000

\*[R/S] is omitted when the printer is present

# User Instructions

				SIZE: 028
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load the program.			
2	Input the matrix A.		[XEQ] EV	a1,1=?
		a <sub>1,1</sub>	[R/S]	a2,1=?
		a <sub>2,1</sub>	[R/S]	a3,1=?
		⋮	⋮	⋮
		a <sub>3,1</sub>	[R/S]	READY
3	View and/or correct the matrix A. While		[B]	a1,1=( )
	viewing any element, if a number is		[R/S]	a2,1=( )
	entered before pressing [R/S] it will		⋮	⋮
	replace the currently displayed element.		[R/S]	READY
4	Print the matrix A. If the printer is		[///] [b]	a1,1=( )
	not attached, this function will act as			⋮
	[B].			READY
5	Calculate the Eigenvalues.		[C]	RLAM.1=( )
	RLAM. ≡ the real portion of $\lambda$ .		[R/S]*	ILAM.1=( )
	ILAM. ≡ the imaginary portion of $\lambda$ .		[R/S]*	RLAM.2=( )
			[R/S]*	ILAM.2=( )
			[R/S]*	LAM.3=( )
6	For a new problem:		[A]	a1,1=?
	This is the same as step 2. Note that			
	the value of each element from the previous			
	matrix A may be reused by just pressing			
	[R/S].			
	*[R/S] is omitted if the printer is attached.			



# Program Listings

01*LBL "EV"		50 ISG 10	
02 SF 21		51 GTO 00	
03 SF 27		52*LBL 02	
04*LBL A		53 "READY"	
05 CF 05	Input A	54 PROMPT	
06 CF 07		55*LBL C	
07 GTO 01		56 RCL 08	
08*LBL b		57 RCL 06	
09 ADV		58 RCL 05	
10 SF 05	Print A	59 RCL 09	Calculate coefficients of 3rd order polynomial
11 SF 07		60 XEQ 09	
12 GTO 01		61 RCL 01	
13*LBL B		62 *	
14 SF 05	View A	63 STO 10	
15 CF 07		64 RCL 02	
16*LBL 01		65 RCL 09	
17 1.003		66 RCL 08	
18 STO 10	Initialize	67 RCL 03	
19 STO 11	Input Routine	68 XEQ 09	
20 1		69 RCL 04	
21 STO 00		70 *	
22*LBL 00		71 ST+ 10	
23 FIX 0		72 RCL 03	
24 CF 29		73 RCL 05	
25 "a"		74 RCL 06	
26 ARCL 11	Input/View/	75 RCL 02	
27 "f,"	Print Routine	76 XEQ 09	
28 ARCL 10		77 RCL 07	
29 "f="		78 *	
30 FC? 05		79 ST+ 10	
31 "f?"		80 RCL 01	
32 FIX 4		81 RCL 05	
33 SF 29		82 RCL 07	
34 RCL IND		83 RCL 03	
00		84 XEQ 09	
35 FS? 05		85 STO 11	
36 ARCL X		86 RCL 01	
37 FC? 07		87 RCL 09	
38 PROMPT		88 RCL 08	
39 FS? 07		89 RCL 06	
40 AVIEW		90 XEQ 09	
41 STO IND		91 ST+ 11	
00		92 RCL 05	
42 ISG 00		93 RCL 09	
43 CLD		94 RCL 04	
44 ISG 11		95 RCL 02	
45 GTO 00		96 XEQ 09	
46 FS? 07		97 ST+ 11	
47 ADV		98 RCL 01	
48 1.003		99 RCL 05	
49 STO 11		100 RCL 09	

# Program Listings

101 +		153 XEQ 05	S+T
102 +		154 RCL 14	
103 CHS		155 RCL 19	
104 STO 12		156 -	
105 CF 06		157 XEQ 05	
106 RCL 10		158 +	
107 X≠0?	Test if constant is zero	159 GTO 04	
108 GTO 10		160♦LBL 03	
109 STO 21		161 RCL 14	
110 RCL 11		162 RCL 13	
111 STO 17		163 3	
112 RCL 12		164 Y↑X	$\lambda_3$ for
113 STO 18		165 CHS	$Q^3+R^2<0$
114 GTO 07		166 SQRT	
115♦LBL 10		167 /	
116 RCL 11		168 ACOS	
117 3		169 3	
118 *		170 /	
119 RCL 12		171 COS	
120 X↑2		172 RCL 13	
121 -		173 CHS	
122 9		174 SQRT	
123 /		175 *	
124 STO 13		176 ST+ X	
125 RCL 11		177♦LBL 04	
126 RCL 12		178 RCL 12	
127 *	Solve for Real Root $\lambda_3$	179 3	
128 9		180 /	$\lambda_3$
129 *		181 -	
130 RCL 10		182 STO 19	
131 27		183 STO 21	
132 *		184 RCL 12	
133 -		185 +	
134 RCL 12		186 STO 18	
135 3		187 RCL 10	
136 Y↑X		188 RCL 19	
137 ST+ X		189 /	
138 -		190 CHS	
139 54		191 STO 17	
140 /		192♦LBL 07	
141 STO 14		193 RCL 18	
142 X↑2		194 X↑2	
143 RCL 13		195 RCL 17	
144 3		196 4	
145 Y↑X		197 *	
146 +		198 -	
147 X<0?		199 CHS	
148 GTO 03		200 X>0?	
149 SQRT		201 SF 06	
150 STO 19		202 ABS	
151 RCL 14		203 SQRT	
152 +			Reduce to second order and calculate $\lambda_1, \lambda_2$

# Program Listings

204 2		255 "I"	
205 /		256 SF 05	
206 STO 20		257 XEQ 08	
207 RCL 18		258 "R"	
208 CHS		259 SF 07	
209 2		260 XEQ 08	
210 /		261 "I"	
211 STO 00		262 SF 07	
212 FS? 06		263 XEQ 08	
213 GTO 11		264 CLA	
214 RCL 20		265 SF 09	
215 +		266 RCL 21	
216 0		267 * LBL 08	
217 X<>Y		268 "FLAM."	
218 RCL 00		269 FS?C 05	
219 RCL 20		270 "F1"	
220 -		271 FS?C 07	
221 0		272 "F2"	
222 X<>Y		273 FS?C 09	
223 GTO 06		274 "F3"	
224 * LBL 11		275 "F="	
225 RCL 20		276 ARCL X	
226 X<>Y		277 AVIEW	
227 RCL 20		278 RDN	
228 CHS		279 END	
229 RCL 00			
230 GTO 06			
231 * LBL 09			
232 *			
233 RDN	ZT-XY		
234 *		80	
235 R↑			
236 -			
237 RTN			
238 * LBL 05			
239 CF 08			
240 X<0?			
241 SF 08	$\pm \sqrt[3]{\quad}$		
242 ABS			
243 3			
244 1/X		90	
245 Y↑X			
246 FS?C 08			
247 CHS			
248 RTN			
249 * LBL 06			
250 CF 09			
251 SF 05	Output routine		
252 CF 07			
253 "R"			
254 XEQ 08		00	



EIGENVALUES FOR 3RD-  
ORDER SYSTEM  
PROGRAM REGISTERS NEEDED: 62

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 1 (1 : 5)



ROW 2 (5 : 12)



ROW 3 (12 : 17)



ROW 4 (18 : 26)



ROW 5 (27 : 31)



ROW 6 (32 : 38)



ROW 7 (39 : 46)



ROW 8 (46 : 52)



ROW 9 (53 : 59)



ROW 10 (60 : 68)



ROW 11 (69 : 78)



ROW 12 (79 : 88)



ROW 13 (89 : 96)



ROW 14 (97 : 107)



ROW 15 (108 : 115)



ROW 16 (116 : 128)



ROW 17 (129 : 139)



ROW 18 (139 : 150)



EIGENVALUES FOR 3RD-  
ORDER SYSTEMHEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 19 (150 : 157)



ROW 20 (158 : 169)



ROW 21 (170 : 181)



ROW 22 (182 : 190)



ROW 23 (191 : 200)



ROW 24 (201 : 210)



ROW 25 (211 : 219)



ROW 26 (220 : 229)



ROW 27 (230 : 240)



ROW 28 (241 : 250)



ROW 29 (251 : 256)



ROW 30 (257 : 262)



ROW 31 (262 : 268)



ROW 32 (268 : 272)



ROW 33 (273 : 279)



ROW 34 (279 : 279)



CHEBYSHEV, LEGENDRE, HERMITE AND LAGUERRE POLYNOMIALS

Label T computes the value of the Chebyshev polynomial  $T_n(x)$  by using the recurrence equation

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

where starting values are  $T_0(x) = 1$ ,  $T_1(x) = x$  and  $n$  is a positive integer.

Label P computes the value of the Legendre polynomial  $P_n(x)$  by using the recurrence equation

$$P_{n+1}(x) = \frac{(2n+1)xP_n(x) - nP_{n-1}(x)}{n+1}$$

where starting values are  $P_0(x)=1$ ,  $P_1(x)=x$  and  $n$  is a positive integer.

Label H computes the value of the Hermite polynomial  $H_n(x)$  by using the recurrence equation

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x)$$

where the starting values are  $H_0(x) = 1$ ,  $H_1(x) = 2x$  and  $n$  is a positive integer.

Label L computes the value of the Laguerre polynomial  $L_n(x)$  by using the recurrence equation

$$L_{n+1}(x) = \frac{(2n+1-x)L_n(x) - nL_{n-1}(x)}{n+1}$$

Note that all four functions leave  $f(x)$  in the  $x$  register if you wish to see more accuracy (see ex. 4).

Examples:

1. Find  $T_3(0.4)$
2. Find  $P_{10}(0.98)$
3. Find  $H_5(3)$
4. Find  $L_6(3)$

## Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 007

[XEQ] [ALPHA] T [ALPHA]

3 [R/S]

.4 [R/S]

[XEQ] [ALPHA] P [ALPHA]

10 [R/S]

0.98 [R/S]

[XEQ] [ALPHA] H [ALPHA]

5 [R/S]

3 [R/S]

[XEQ] [ALPHA] L [ALPHA]

6 [R/S]

3 [R/S]

■ [FIX] 4

## Display:

N?

X?

T3(0.40)=-0.94

N?

X?

P10(0.98)=0.16

N?

X?

H5(3.00)=3,816.00

N?

X?

L6(3.00)=-0.01

-0.0125





# Program Listings

01 *LBL "T"	Chebyshev polynomial	52 STO 03	
02 XEQ A		53 X<>Y	
03 STO 00		54 X<=Y?	
04 2		55 GTO 04	
05 STO 01		56 2	
06 *		57 STO 02	
07 STO 02		58 *LBL 03	----- Loop for recur- rence equation
08 CLX		59 RCL 04	
09 X<>Y		60 RCL 01	
10 X<=Y?		61 *	
11 GTO 00		62 ENTER↑	
12 1		63 ENTER↑	
13 STO 03		64 RCL 03	
14 X=Y?		65 -	
15 GTO 01		66 +	
16 *LBL 02	----- Loop for recur- rence equation	67 LASTX	
17 CLX		68 RCL 04	
18 RCL 02		69 STO 03	
19 RCL 00		70 CLX	
20 *		71 RCL 02	
21 RCL 03		72 /	
22 LASTX		73 -	
23 STO 03		74 STO 04	
24 RDN		75 RCL 00	
25 -		76 RCL 02	
26 STO 00		77 1	
27 CLX		78 +	
28 RCL 01		79 STO 02	----- Test
29 1		80 X<=Y?	
30 +		81 GTO 03	
31 STO 01		82 RCL 04	
32 X<=Y?		83 "P"	
33 GTO 02	Test, $r_1 \leq n_1$ so back to loop	84 GTO d	
34 *LBL 01		85 *LBL 04	
35 RCL 00		86 RCL 01	
36 "T"		87 RTN	----- Hermite polynomial
37 GTO d		88 *LBL H	
38 *LBL 00		89 XEQ A	
39 1		90 STO 01	
40 "T"		91 2	
41 GTO d		92 *	
42 *LBL "P"	----- Legendre polynomial	93 STO 03	
43 XEQ A		94 CLX	
44 STO 01		95 X<>Y	
45 STO 04		96 X<=Y?	
46 CLX		97 GTO 00	
47 X<>Y		98 1	
48 STO 00		99 STO 00	
49 X<=Y?		100 STO 02	
50 GTO 00		101 X<>Y	
51 1		102 X<=Y?	

# Program Listings

103 GTO 05		154 RCL 03	
104 * LBL 06	-----	155 -	
105 RCL 00	Test for recur-	156 RCL 04	
106 RCL 03	rence equation	157 +	
107 STO 00		158 RCL 01	
108 RCL 01		159 RCL 04	
109 *		160 *	
110 X<>Y		161 RCL 03	
111 RCL 02		162 -	
112 *		163 RCL 02	
113 -		164 /	
114 2		165 RCL 04	
115 *		166 STO 03	
116 STO 03		167 RDN	
117 CLX		168 -	
118 RCL 02		169 STO 04	
119 1		170 RCL 00	
120 +		171 RCL 02	
121 STO 02		172 1	
122 X<>Y	-----	173 +	
123 X≠Y?	Test	174 STO 02	
124 GTO 06		175 X<=Y?	-----
125 RCL 03		176 GTO 08	Test
126 "H"		177 * LBL 07	
127 GTO d		178 RCL 04	
128 * LBL 05		179 "L"	
129 RCL 03		180 GTO d	
130 RTN	-----	181 STOP	-----
131 * LBL "L"	Laguerre	182 * LBL A	Input n and x
132 XEQ A	polynomial	183 "N?"	
133 ENTER↑		184 PROMPT	
134 1		185 STO 05	
135 STO 03		186 "X?"	
136 +		187 PROMPT	
137 STO 01		188 STO 06	
138 2		189 RTN	-----
139 STO 02		190 * LBL d	Display
140 X<>Y		191 CF 29	
141 -		192 FIX 0	
142 STO 04		193 ARCL 05	
143 CLX		194 "F<"	
144 X<>Y		195 SF 29	
145 X<=Y?		196 FIX 2	
146 GTO 00		197 ARCL 06	
147 STO 00		198 "F>="	
148 1		199 ARCL X	
149 X<>Y	-----	200 RVIEW	
150 X<=Y?	Loop for recur-	201 STOP	
151 GTO 07	rence equation	202 .END.	
152 * LBL 08			
153 RCL 04			

# REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS			STATUS					
00	temp. change	50	SIZE	007	TOT. REG.	45	USER MODE	
	temp. change		ENG		FIX	2	SCI	
	temp. change		DEG		RAD		GRAD	
	temp. change						ON	OFF X
05	n	55	FLAGS					
	x		#	INIT S/C	SET INDICATES	CLEAR INDICATES		
					NONE			
10		60						
15		65						
20		70						
25		75						
30		80						
35		85						
			ASSIGNMENTS					
			FUNCTION	KEY	FUNCTION	KEY		
40		90	NONE					
45		95						

CHEBYSHEV LEGENDRE HERMITE  
AND LAGUERRE POLYNOMIALS  
PROGRAM REGISTERS NEEDED: 39

ROW 1 (1 - 7)



ROW 2 (8 - 18)



ROW 3 (19 - 31)



ROW 4 (32 - 40)



ROW 5 (40 - 44)



ROW 6 (45 - 55)



ROW 7 (56 - 68)



ROW 8 (69 - 81)



ROW 9 (81 - 89)



ROW 10 (89 - 99)



ROW 11 (100 - 111)



ROW 12 (112 - 124)



ROW 13 (124 - 131)



ROW 14 (131 - 140)



ROW 15 (141 - 151)



ROW 16 (152 - 164)



ROW 17 (165 - 176)



ROW 18 (177 - 183)



CHEBYSHEV LEGENDRE HERMITE  
AND LAGUERRE POLYNOMIALS

ROW 19 (184 - 192)



ROW 20 (192 - 198)



ROW 21 (198 - 202)



## SIXTEEN-POINT GAUSSIAN QUADRATURE

This program will compute approximations for integrals over finite or infinite intervals by the sixteen-point Gauss-Legendre quadrature method. If  $f(x)$  is the function to be integrated, then either

$$\int_a^b f(x) \, dx \quad \text{or} \quad \int_a^\infty f(x) \, dx \quad \text{may be found.}$$

The function  $f(x)$  must be explicitly known and keyed into program memory under a separate program label. This function assumes the value of  $x$  will be in the X-register. Registers greater than 20 and the stack are available to the user to define  $f(x)$ .

$$\int_a^b f(x) \, dx = \frac{b-a}{2} \sum_{i=1}^{16} W_i f\left(\frac{Z_i(b-a) + b + a}{2}\right)$$

$$\int_a^\infty f(x) \, dx = 2 \sum_{i=1}^{16} \frac{W_i}{(1 + Z_i)^2} f\left(\frac{2}{1+Z_i} + a-1\right)$$

The constants ( $W_i$ 's and  $Z_i$ 's) can be stored on a data card; their values and memory locations are given on the following page.

## NOTE:

1. The Trig mode should be set to radians by your routine to avoid any oversights in computing integrals involving trig functions.
2. The total space required to execute this program is 46 registers plus the program space used to define  $f(x)$ .

## Examples:

1. Find  $\int_1^4 1/X_2 dx$
2. Find  $\int_1^\infty 1/X_2 dx$

## Keystrokes:

```

[XEQ] [ALPHA] SIZE [ALPHA] 021
[////] [FIX] 4
2.715245941 [EEX] 2 [CHS] [STO] 01
9.894009350 [EEX] 1 [CHS] [STO] 02
6.225352394 [EEX] 2 [CHS] [STO] 03
9.445750231 [EEX] 1 [CHS] [STO] 04
9.515851168 [EEX] 2 [CHS] [STO] 05
8.656312024 [EEX] 1 [CHS] [STO] 06
1.246289713 [EEX] 1 [CHS] [STO] 07
7.554044084 [EEX] 1 [CHS] [STO] 08
1.495959888 [EEX] 1 [CHS] [STO] 09
6.178762444 [EEX] 1 [CHS] [STO] 10
1.691565194 [EEX] 1 [CHS] [STO] 11
4.580167777 [EEX] 1 [CHS] [STO] 12
1.826034150 [EEX] 1 [CHS] [STO] 13
2.816035508 [EEX] 1 [CHS] [STO] 14
1.894506105 [EEX] 1 [CHS] [STO] 15
9.501250984 [EEX] 2 [CHS] [STO] 16
[////] [GTO] ..
[PRGM]
[////] [LBL] [ALPHA] X [ALPHA]
[1/x]
[////] [x2]
[////] [GTO] ..
[PRGM]
[XEQ] [ALPHA] GAUSS [ALPHA]
[A]
X [R/S]
1 [R/S]
4 [R/S]
[B]
X [R/S]
1 [R/S]

```

## Display:

```

01 LBL 'X
02 1/X
03 X^2

NAME?
a?
b?
0.7500
NAME?
a?
1.0000

```





# Program Listings

01♦LBL "GAU SS"	Set User Mode	51♦LBL 04	
02 SF 27		52 RCL 00	
03 STOP		53 2	2 $\Sigma$
04♦LBL A	from a to b	54 *	
05 CF 05		55 RTN	
06 GTO 00		56♦LBL 05	
07♦LBL B	from a to $\infty$	57 RCL IND	
08 SF 05		19	a to b
09♦LBL 00		58 FS? 06	
10 "NAME?"		59 CHS	Calculation
11 AON		60 RCL 18	
12 STOP	Input	61 RCL 17	Subroutine
13 AOFF		62 -	
14 ASTO 20		63 *	
15 "a?"		64 RCL 18	
16 PROMPT		65 +	
17 STO 17		66 RCL 17	
18 "b?"		67 +	
19 FC? 05		68 2	
20 PROMPT		69 /	
21 FC? 05		70 XEQ IND	
22 STO 18		20	
23 0		71 DSE 19	
24 STO 00		72 RCL IND	
25 SF 06		19	
26♦LBL 01		73 *	
27 16		74 ST+ 00	
28 STO 19		75 RTN	
29♦LBL 11		76♦LBL 06	
30 FS? 05		77 RCL IND	
31 XEQ 06		19	
32 FC? 05		78 FS? 06	
33 XEQ 05	Iterative loop	79 CHS	
34 DSE 19		80 1	a to $\infty$
35 GTO 11		81 +	
36 FC?C 06		82 2	Calculation
37 SF 06		83 X<>Y	
38 FC? 06		84 /	Subroutine
39 GTO 01		85 RCL 17	
40 FS? 05		86 +	
41 GTO 04		87 1	
42♦LBL 02		88 -	
43 RCL 18		89 XEQ IND	
44 RCL 17		20	
45 -	$\frac{b-a}{2} \Sigma$	90 RCL IND	
46 2		19	
47 /		91 FS? 06	
48 RCL 00		92 CHS	
49 *		93 1	
50 RTN		94 +	
		95 X $\uparrow$ 2	





SIXTEEN-POINT GAUSSIAN  
QUADRATURE  
PROGRAM REGISTERS NEEDED: 24

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 1 (1 : 3)



ROW 2 (4 : 8)



ROW 3 (9 : 16)



ROW 4 (16 : 24)



ROW 5 (25 : 31)



ROW 6 (31 : 37)



ROW 7 (37 : 45)



ROW 8 (46 : 56)



ROW 9 (57 : 65)



ROW 10 (66 : 74)



ROW 11 (75 : 84)



ROW 12 (85 : 94)



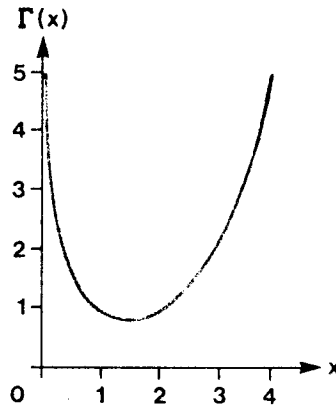
ROW 13 (94 : 100)



## GAMMA FUNCTION

This program approximates the value of the gamma function,  $\Gamma(x)$ , for  $1 \leq x \leq 70$ .

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt$$



1.  $\Gamma(x) = (x - 1) \Gamma(x-1)$  if  $x > 2$
2. For  $1 \leq x \leq 2$ , polynomial approximation can be used.  

$$\Gamma(x) \cong 1 + b_1 (x - 1) + b_2 (x - 1)^2 + \dots + b_8 (x - 1)^8$$

where

$b_1 = -0.577191652,$	$b_2 = 0.988205891$
$b_3 = -0.897056937,$	$b_4 = 0.918206857$
$b_5 = -0.756704078,$	$b_6 = 0.482199394$
$b_7 = -0.193527818,$	$b_8 = 0.035868343$

### Remarks:

1. This program can be used to find the generalized factorial  $x!$  for  $0 \leq x \leq 69$ . where  $x! = \Gamma(x + 1)$ .
2. When the value keyed in for  $x$  is an integer,  $\Gamma(x)$  is evaluated as the factorial of  $(x-1)$ .
3. If  $x < 1$ , the program will halt and display "ILLEGAL X".

### References:

Handbook of Mathematical Functions, Abramowitz and Stegun, National Bureau of Standards, 1968.

Examples: Find the gamma function for the arguments

5.25, 8, and 3.34.

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 001

[////] [FIX] 2

5.25 [XEQ] [ALPHA] GAMMA [ALPHA]

8 [R/S]

3.34 [R/S]

Display:

GAMMA=35.21

GAMMA=5,040.00

GAMMA=2.80







# REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS			STATUS					
00	Γ	50	SIZE	001	TOT. REG.	29	USER MODE	
			ENG	_____	FIX	_____	ON	OFF
			DEG	_____	RAD	_____	GRAD	_____
05		55	FLAGS					
			#	INIT S/C	SET INDICATES	CLEAR INDICATES		
10		60						
15		65						
20		70						
25		75						
30		80						
35		85						
			ASSIGNMENTS					
			FUNCTION	KEY	FUNCTION	KEY		
40		90						
45		95						

## GAMMA FUNCTION

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH ORDER MATH

PROGRAM REGISTERS NEEDED: 24

ROW 1 (1 : 4)



ROW 2 (4 : 8)



ROW 3 (9 : 18)



ROW 4 (19 : 25)



ROW 5 (25 : 27)



ROW 6 (27 : 30)



ROW 7 (30 : 33)



ROW 8 (34 : 37)



ROW 9 (38 : 41)



ROW 10 (42 : 45)



ROW 11 (45 : 49)



ROW 12 (50 : 56)



ROW 13 (56 : 60)



## BESSEL FUNCTIONS, ERROR FUNCTION

The first routine computes the Bessel functions  $J_n(x)$  and  $I_n(x)$ , where  $n$  is a positive integer and  $x > 0$ . The second of the two routines finds the error function and complementary error function for positive arguments.

### Bessel Functions

The Bessel functions  $J_n(x)$  and  $I_n(x)$  are computed by generating trial values  $T_k$  through the use of recurrence relations. The recurrence is begun at an index  $m$  given by

$$m = 2 \text{ INT} \left[ \frac{6 + \max(n, z) + \frac{9z}{z+2}}{2} \right]$$

where

$$z = \frac{3x}{2} .$$

The initial values selected for recurrence are  $T_{m+1} = 10^{-9}$ ,  $T_{m+2} = 0$ .

For the functions  $J_n(x)$ , each term  $T_k$  ( $0 \leq k \leq m$ ) is computed by the relation

$$T_k(x) = \frac{2(k+1)}{x} T_{k+1}(x) - T_{k+2}(x)$$

beginning with  $k = m$ .

$J_n(x)$  is then found by dividing the term  $T_n(x)$  by the normalizing constant

$$K = T_0(x) + 2 \sum_{k=1}^{m/2} T_{2k}(x).$$

After calculating a  $J_n(x)$ , the values of  $J_0(x)$  and  $J_1(x)$  may also be found with very little additional computation.

For the functions  $I_n(x)$ , each  $T_k$  is calculated from the recurrence relation

$$T_k(x) = \frac{2(k+1)}{x} T_{k+1}(x) + T_{k+2}(x),$$

$0 \leq k \leq m$ , beginning with  $k = m$ .

$I_n(x)$  is then found from the equation:

$$I_n(x) = e^x \frac{T_n(x)}{T_0(x) + 2 \sum_{k=1}^m T_k(x)}$$

Error Function

The error function is defined as

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

and the complementary error function as

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x).$$

For large values of  $x$  ( $\geq 3$ ), the error function is very close to 1. If  $\operatorname{erfc}(x)$  is computed as  $1 - \operatorname{erf}(x)$ , most of the significant figures of  $\operatorname{erfc}(x)$  will be lost for  $x > 3$ . Hence two different algorithms are employed in this program, one for  $x \leq 3$  and one for  $x > 3$ . For  $x \leq 3$ , the error function is computed by a series sum

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \sum_{n=0}^{\infty} \frac{2^n}{1 \cdot 3 \cdot \dots \cdot (2n+1)} x^{2n+1}$$

and the complementary error function

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x).$$

For  $x > 3$ , the complementary error function is computed first, by the asymptotic expansion

$$\operatorname{erfc}(x) = \frac{1}{x \sqrt{\pi}} e^{-x^2} \left[ 1 + \sum_{n=1}^{\infty} \frac{(-1)^n 1 \cdot 3 \cdot \dots \cdot (2n-1)}{(2x^2)^n} \right]$$

and the error function by

$$\operatorname{erf}(x) = 1 - \operatorname{erfc}(x).$$

The accuracy of the calculation of  $\operatorname{erf}(x)$  and  $\operatorname{erfc}(x)$  from series sums may be controlled by the user. For  $x \leq 3$ , it is quite reasonable to specify 9 for maximum accuracy; for  $x > 3$ , the series may never converge with 9, and a safer specification would be 6 digits.

## Remarks:

1. The range of values  $0 \leq x \leq 10^{-6}$  is out of bounds for the Bessel functions in this program. In this range, however, one may take  $J_0(x) = J_0(0) = I_0(x) = I_0(0) = 1$ , and  $J_n(x) = J_n(0) = I_n(x) = I_n(0) = 0$ ,  $n \neq 0$ .
2. The computation of  $\text{erfc}(x)$  will halt on overflow for  $x \geq 15$ .

Reference: Abramowitz and Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards, 1968.

## Examples:

1. Find  $J_5$  (9.2)
2. Find  $J_0$  (9.2)
3. Find  $J_1$  (9.2)
4. Find  $I_3$  (4.7)
5. Find  $\text{erf}$  and  $\text{erfc}$  4.55 to 6 places

Keystrokes:	Display:	
[USER]		(set USER mode)
[XEQ] [ALPHA] SIZE [ALPHA] 007		
[XEQ] [ALPHA] INIT [ALPHA]		
[J]	N?	
5 [R/S]	X?	
1) 9.2 [R/S]	J=-0.1005	
2) [R/S]	J0=-0.1367	
3) [R/S]	J1=0.2174	
[I]	N?	
3 [R/S]	X?	
4) 4.7 [R/S]	I=7.4195	
[E]	ACCURACY?	
6 [R/S]	X?	
5) 4.55 [R/S]	ERF=1.000000	
[R/S]	ERFC=1.237405E-10	



# Program Listings

01*LBL "INI T"	"J"	51 +	
02 CLRG		52 2	
03 RTN		53 +	
04*LBL J		54 STO 06	
05 XEQ a		55 3	
06 SF 00		56 RCL 03	
07*LBL 09		57 /	
08 XEQ b		58 STO 02	
09 CF 02		59 0	
10 ST+ 00		60 STO 05	
11 XEQ b		61 STO 00	
12 FS?C 02		62 E-9	
13 GTD 09		63 STO 04	
14 RCL 03		64 RTN	
15 RCL 00		65*LBL b	
16 ENTER↑		66 DSE 06	----- Compute one term
17 +		67 SF 02	FZ set except
18 RCL 05		68 RCL 06	for F = 0
19 -		69 RCL 01	
20 /		70 X*Y?	
21 "J="		71 GTD 00	
22 XEQ d		72 RCL 04	
23 GTD C	----- input n + x	73 STO 03	
24*LBL a		74*LBL 00	
25 FIX 4		75 RDN	
26 "N?"		76 RCL 05	
27 PROMPT		77 FS? 00	
28 STO 01		78 CHS	
29 "X?"		79 X<>Y	
30 PROMPT	----- initialization for Bessel ( $J_n + I_n$ )	80 RCL 02	
31 1.5		81 *	
32 *		82 RCL 04	
33 STO 03		83 STO 05	
34 RCL 01		84 *	
35 X<=Y?		85 +	
36 X<>Y		86 STO 04	
37 6		87 RTN	
38 +		88*LBL C	----- Compute $J_0(x) +$ $J_1(x)$
39 RCL 03		89 RCL 05	
40 9		90 RCL 00	
41 *		91 ENTER↑	
42 RCL 03		92 +	
43 2		93 RCL 05	
44 +		94 -	
45 /		95 /	
46 +		96 "J0="	
47 2		97 XEQ d	
48 /		98 RCL 04	
49 INT		99 CHS	
50 ENTER↑		100 RCL 00	
		101 ENTER↑	
		102 +	



# Program Listings

```

103 RCL 05
104 -
105 /
106 "J1="
107 GTO d
108 LBL I
109 CF 00
110 XEQ a
111 LBL 08
112 ST+ 00
113 XEQ b
114 FS?C 02
115 GTO 08
116 RCL 03
117 RCL 00
118 ENTER↑
119 +
120 RCL 05
121 -
122 /
123 2
124 RCL 02
125 /
126 E↑X
127 *
128 "I="
129 GTO d
130 LBL E
131 "ACCURAC
Y?"
132 PROMPT
133 FIX IND
X
134 "X?"
135 PROMPT
136 STO 01
137 X↑2
138 STO 04
139 2
140 *
141 STO 02
142 1
143 STO 03
144 RCL 04
145 E↑X
146 PI
147 SQRT
148 *
149 STO 04
150 3
151 RCL 01

```

-----  
"I"

-----  
Compute error

```

152 X>Y?
153 GTO 03
154 LBL 07
155 RCL 02
156 RCL 03
157 2
158 +
159 STO 03
160 /
161 RCL 01
162 *
163 STO 01
164 +
165 X<>Y
166 RND
167 X<>Y
168 RND
169 X=Y?
170 GTO 00
171 LASTX
172 GTO 07
173 LBL 00
174 LASTX
175 RCL 04
176 /
177 2
178 *
179 1
180 X<>Y
181 -
182 LASTX
183 GTO 02
184 LBL 03
185 RCL 02
186 1/X
187 STO 02
188 RCL 01
189 1/X
190 STO 01
191 LBL 06
192 RCL 02
193 RCL 03
194 2
195 -
196 STO 03
197 *
198 RCL 01
199 *
200 STO 01
201 +
202 X<>Y
203 RND

```

-----  
Loop for erf

-----  
Exit erf

-----  
Find erfc  
x > 3

-----  
Loop for erfc

# Program Listings

204 X<>Y		51	
205 RND			
206 X=Y?			
207 GTO 00			
208 LASTX			
209 GTO 06			
210♦LBL 00	-----		
211 LASTX	erfc(x)		
212 RCL 04			
213 /			
214 1		60	
215 X<>Y			
216 -	-----		
217 LASTX	erf(x)		
218 X<>Y			
219♦LBL 02			
220 "ERF="			
221 XEQ d			
222 X<>Y			
223 "ERFC="	-----		
224♦LBL d	display	70	
225 ARCL X			
226 AVIEW			
227 STOP			
228 .END.			
30		80	
40		90	
50		00	

# REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS			STATUS				
00	$\Sigma T_k$	50	SIZE	007	TOT. REG.	53	USER MODE
	n; erf term		ENG		FIX	09	SCI
	2/x		DEG		RAD		GRAD
	1.5x, T <sub>n</sub>		<b>FLAGS</b> #    INIT S/C    SET INDICATES    CLEAR INDICATES				
	T <sub>k</sub> ; $(e^{x^2} \sqrt{\pi})^{-1}$						
05	T <sub>k+1</sub>	55					
	K; places						
10		60					
15		65					
20		70					
25		75					
30		80					
35		85					
			<b>ASSIGNMENTS</b> FUNCTION    KEY    FUNCTION    KEY				
40		90					
45		95					

BESSEL FUNCTIONS  
ERROR FUNCTION  
PROGRAM REGISTERS NEEDED: 47

ROW 1 (1 - 5)



ROW 2 (5 - 11)



ROW 3 (11 - 20)



ROW 4 (21 - 25)



ROW 5 (26 - 32)



ROW 6 (33 - 45)



ROW 7 (46 - 58)



ROW 8 (59 - 67)



ROW 9 (67 - 77)



ROW 10 (78 - 89)



ROW 11 (90 - 97)



ROW 12 (98 - 107)



ROW 13 (107 - 113)



ROW 14 (113 - 122)



ROW 15 (123 - 130)



ROW 16 (131 - 133)



ROW 17 (134 - 144)



ROW 18 (145 - 156)



BESSEL FUNCTIONS  
ERROR FUNCTION

ROW 19 (157 - 169)



ROW 20 (170 - 180)



ROW 21 (181 - 192)



ROW 22 (193 - 205)



ROW 23 (206 - 216)



ROW 24 (217 - 223)



ROW 25 (223 - 228)



ROW 26 (228 - 228)



## CHARACTERISTIC EQUATION OF A 4 X 4 MATRIX

(THIS PROGRAM REQUIRES ONE ADDITIONAL MEMORY MODULE)

Given

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} \end{pmatrix}$$

The characteristic equation is  $\lambda^4 + r_1\lambda^3 + r_2\lambda^2 + r_3\lambda + r_4 = 0$

where

$$r_1 = -(a_{1,1} + a_{2,2} + a_{3,3} + a_{4,4})$$

$$r_2 = (a_{1,1} + a_{3,3}) a_{2,2} + (a_{1,1} + a_{4,4}) a_{3,3} + (a_{1,1} + a_{2,2}) a_{4,4} - a_{2,4} a_{4,2} - a_{3,4} a_{4,3} - a_{2,3} a_{3,2} - a_{1,2} a_{2,1} - a_{1,3} a_{3,1} - a_{1,4} a_{4,1}$$

$$r_3 = -\det(A_1) - a_{1,1}(a_{2,2} a_{3,3} + a_{2,2} a_{4,4} + a_{3,3} a_{4,4} - a_{2,4} a_{4,2} - a_{3,4} a_{4,3} - a_{2,3} a_{3,2}) + a_{1,2}[a_{2,1}(a_{3,3} + a_{4,4}) - a_{2,4} a_{4,1} - a_{2,3} a_{3,1}] - a_{1,3}[-a_{3,1}(a_{2,2} + a_{4,4}) + a_{2,1} a_{3,2} + a_{3,4} a_{4,1}] + a_{1,4}[a_{4,1}(a_{2,2} + a_{3,3}) - a_{3,1} a_{4,3} - a_{2,1} a_{4,2}]$$

$$r_4 = a_1 \det(A_1) - a_2 \det(A_2) + a_3 \det(A_3) - a_4 \det(A_4)$$

and

$$A_1 = \begin{pmatrix} a_{2,2} & a_{2,3} & a_{2,4} \\ a_{3,2} & a_{3,3} & a_{3,4} \\ a_{4,2} & a_{4,3} & a_{4,4} \end{pmatrix} \quad A_2 = \begin{pmatrix} a_{2,1} & a_{2,3} & a_{2,4} \\ a_{3,1} & a_{3,3} & a_{3,4} \\ a_{4,1} & a_{4,3} & a_{4,4} \end{pmatrix} \quad A_3 = \begin{pmatrix} a_{2,1} & a_{2,2} & a_{2,4} \\ a_{3,1} & a_{3,2} & a_{3,4} \\ a_{4,1} & a_{4,2} & a_{4,4} \end{pmatrix}$$

and

$$A_4 = \begin{pmatrix} a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{pmatrix}$$

NOTE: Trace (A) =  $r_1$ , det (A) =  $-r_4$ .

Example: Find the characteristic equation of the matrix

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 2 & -1 \\ 3 & -1 & 0 & 2 \\ -2 & -1 & -1 & 0 \end{pmatrix}$$

(Ans.  $\lambda^4 - \lambda^3 + 7\lambda + 2 = 0$ )

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 026

[XEQ] [ALPHA] CEM [ALPHA]

1 [R/S]

0 [R/S]

1 [R/S]

0 [R/S]

1 [R/S]

0 [R/S]

2 [R/S]

1 [CHS] [R/S]

3 [R/S]

1 [CHS] [R/S]

0 [R/S]

2 [R/S]

2 [CHS] [R/S]

1 [CHS] [R/S]

1 [CHS] [R/S]

0 [R/S]

[R/S]\*

[R/S]\*

[R/S]\*

Display:

a<sub>1,1</sub>=?

a<sub>1,2</sub>=?

a<sub>1,3</sub>=?

a<sub>1,4</sub>=?

a<sub>2,1</sub>=?

a<sub>2,2</sub>=?

a<sub>2,3</sub>=?

a<sub>2,4</sub>=?

a<sub>3,1</sub>=?

a<sub>3,2</sub>=?

a<sub>3,3</sub>=?

a<sub>3,4</sub>=?

a<sub>4,1</sub>=?

a<sub>4,2</sub>=?

a<sub>4,3</sub>=?

a<sub>4,4</sub>=?

R1=-1.000

R2=0.000

R3=7.000

R4=2.000

\*[R/S] is omitted if printer is attached.





# Program Listings

01*LBL "CEM"		49 STO 02	
02 SF 21	Initialize	50 ISG 01	
03 SF 27		51 GTO 04	
04*LBL A		52 "READY"	
05 CF 05		53 PROMPT	
06 CF 07	Input matrix	54*LBL C	Calculation
07 GTO 03		55 RCL 10	of r <sub>1</sub>
08*LBL b		56 STO 00	
09 ADV		57 RCL 15	
10 SF 05	Print matrix	58 +	
11 SF 07		59 RCL 20	
12 GTO 03		60 +	
13*LBL B		61 RCL 25	
14 SF 05	View matrix	62 +	
15 CF 07		63 CHS	
16*LBL 03		64 "R1"	
17 1.004		65 XEQ 02	
18 STO 01	Initialize	66 RCL 10	Calculation
19 STO 02	Input Routine	67 RCL 20	of r <sub>2</sub>
20 10		68 +	
21 STO 00		69 RCL 15	
22*LBL 04		70 *	
23 FIX 0		71 RCL 10	
24 CF 29		72 RCL 25	
25 "a"		73 +	
26 ARCL 01	Common Input	74 RCL 20	
27 "f,"	Routine	75 *	
28 ARCL 02		76 +	
29 "f="		77 RCL 10	
30 FC? 05		78 RCL 15	
31 "f?"		79 +	
32 FIX 4		80 RCL 25	
33 SF 29		81 *	
34 RCL IND		82 +	
00		83 RCL 17	
35 FS? 05		84 RCL 23	
36 ARCL X		85 *	
37 FC? 07		86 -	
38 PROMPT		87 RCL 21	
39 FS? 07		88 RCL 24	
40 AVIEW		89 *	
41 STO IND		90 -	
00		91 RCL 16	
42 ISG 00		92 RCL 19	
43 CLD		93 *	
44 ISG 02		94 -	
45 GTO 04		95 RCL 11	
46 FS? 07		96 RCL 14	
47 ADV		97 *	
48 1.004		98 -	
		99 RCL 12	

# Program Listings

100 RCL 18		151 RCL 10	
101 *		152 *	
102 -		153 -	
103 RCL 13		154 RCL 20	
104 RCL 22		155 RCL 25	
105 *		156 +	
106 -		157 RCL 14	
107 "R2"		158 *	
108 XEQ 02	Calculate r <sub>3</sub>	159 RCL 17	
109 RCL 15		160 RCL 21	
110 STO 01		161 *	
111 RCL 16		162 -	
112 STO 02		163 RCL 16	
113 RCL 17		164 RCL 18	
114 STO 03		165 *	
115 RCL 19		166 -	
116 STO 04		167 RCL 11	
117 RCL 20		168 *	
118 STO 05		169 +	
119 RCL 21		170 RCL 15	
120 STO 06		171 RCL 25	
121 RCL 23		172 +	
122 STO 07		173 RCL 18	
123 RCL 24		174 *	
124 STO 08		175 RCL 19	
125 RCL 25		176 RCL 14	
126 STO 09		177 *	
127 XEQ 00		178 -	
128 ST* 00		179 RCL 21	
129 CHS		180 RCL 22	
130 RCL 25		181 *	
131 RCL 20		182 -	
132 +		183 RCL 12	
133 LASTX		184 *	
134 *		185 +	
135 LASTX		186 RCL 15	
136 RCL 25		187 RCL 20	
137 *		188 +	
138 +		189 RCL 22	
139 RCL 17		190 *	
140 RCL 23		191 RCL 18	
141 *		192 RCL 24	
142 -		193 *	
143 RCL 21		194 -	
144 RCL 24		195 RCL 14	
145 *		196 RCL 23	
146 -		197 *	
147 RCL 16		198 -	
148 RCL 19		199 RCL 13	
149 *		200 *	
150 -		201 +	

# Program Listings

202 "R3"		253 *	
203 XEQ 02		254 RCL 04	
204 RCL 14	Calculate r <sub>4</sub>	255 RCL 09	
205 STO 01		256 *	
206 RCL 18		257 -	
207 STO 04		258 RCL 02	
208 RCL 22		259 *	
209 STO 07		260 +	
210 XEQ 00		261 RCL 04	
211 RCL 11		262 RCL 08	
212 *		263 *	
213 ST- 00		264 RCL 05	
214 RCL 15		265 RCL 07	
215 STO 02		266 *	
216 RCL 19		267 -	
217 STO 05		268 RCL 03	
218 RCL 23		269 *	
219 STO 08		270 +	
220 XEQ 00		271 .END.	
221 RCL 12			70
222 *			
223 ST+ 00			
224 RCL 16			
225 STO 03			
226 RCL 20			
227 STO 06			
228 RCL 24			
229 STO 09			
230 XEQ 00			
231 RCL 13			
232 *			80
233 ST- 00			
234 RCL 00			
235 "R4"			
236 LBL 02			
237 "F="			
238 ARCL X	Output routine		
239 AVIEW			
240 RTN			
241 LBL 00			
242 RCL 05			
243 RCL 09			90
244 *	Find		
245 RCL 06	R <sub>1</sub> R <sub>2</sub> R <sub>3</sub>		
246 RCL 08	R <sub>4</sub> R <sub>5</sub> R <sub>6</sub>		
247 *	R <sub>7</sub> R <sub>8</sub> R <sub>9</sub>		
248 -			
249 RCL 01			
250 *			
251 RCL 06			
252 RCL 07			
			00



CHARACTERISTIC EQUATION  
OF A 4 X 4 MATRIX  
PROGRAM REGISTERS NEEDED: 60

HEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 1 (1 : 4)



ROW 2 (5 : 11)



ROW 3 (12 : 17)



ROW 4 (17 : 25)



ROW 5 (26 : 31)



ROW 6 (31 : 37)



ROW 7 (37 : 45)



ROW 8 (45 : 51)



ROW 9 (51 : 57)



ROW 10 (58 : 65)



ROW 11 (65 : 74)



ROW 12 (75 : 84)



ROW 13 (85 : 93)



ROW 14 (94 : 104)



ROW 15 (105 : 112)



ROW 16 (113 : 121)



ROW 17 (121 : 128)



ROW 18 (129 : 138)



CHARACTERISTIC EQUATION  
OF A 4 X 4 MATRIXHEWLETT PACKARD  
SOLUTION BOOK:  
HIGH-LEVEL MATH

ROW 19 (139 : 147)



ROW 20 (147 : 156)



ROW 21 (157 : 165)



ROW 22 (166 : 175)



ROW 23 (176 : 186)



ROW 24 (187 : 195)



ROW 25 (196 : 203)



ROW 26 (204 : 212)



ROW 27 (213 : 220)



ROW 28 (221 : 229)



ROW 29 (230 : 237)



ROW 30 (237 : 247)



ROW 31 (248 : 260)



ROW 32 (261 : 271)



## 4 x 4 MATRIX OPERATIONS

(THIS PROGRAM REQUIRES TWO ADDITIONAL MEMORY MODULES)

This program allows the calculations of the determinant, and inverse of a 4 x 4 matrix, and the solution of a system of simultaneous equations in 4 unknowns.

The method used in this program is that of Gaussian elimination with partial pivoting. Space does not allow a full treatment of the pertinent equations; however, the Comments section of the program listing shows the operations in detail, step by step.

Basically, the program allows for input of the matrix A and transforms A into an upper triangular matrix U, assuming A is nonsingular. The multipliers used to accomplish this transformation form a lower triangular matrix, L, which has 1's along its diagonal. If we disregard pivoting (a technique of row interchanges which may improve accuracy and which may introduce one or more permutation matrices) then the relationship among these matrices is  $U = LA$ . The original matrix A will be lost. The initial elements  $a_{ij}$  have been replaced by the elements of U ( $i \leq j$ ) and of L ( $i > j$ ). (The elements of U will still be referred to as  $a_{ij}$ ; those of L will be called  $m_{ij}$  in the program listing comments). The second part of the program uses the transformed matrices U and L to compute the determinant and inverse of A, and to solve systems of simultaneous equations.

Equations:

$$\text{Let } A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

The determinant of A, Det A, is found after its transformation to U by the product of the diagonal elements:

$$\text{Det } A = (-1)^k a_{11} a_{22} a_{33} a_{44},$$

where k is the number of row interchanges required by pivoting.

A set of 4 simultaneous equations in 4 unknowns may be written as

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 = b_1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 = b_2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 = b_3$$

$$a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 = b_4$$

where the  $\{x_i\}$  are unknowns and the  $\{b_i\}$  constants.

In matrix notation, this becomes  $Ax = b$ , where  $x$  and  $b$  are the column

vectors  $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$  and  $\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$  respectively.

This problem is solved (neglecting pivoting) as  $Ux = Lb$ .

Let  $C$  be the inverse of  $A$ , i.e., the  $4 \times 4$  matrix such that  $AC = CA = I$ , where  $I$  is the  $4 \times 4$  identity matrix.

$C$  is computed a column at a time in the following way:

let  $c^{(j)}$  be the  $j^{\text{th}}$  column vector of  $C$ , i.e.,

$$c^{(j)} = \begin{bmatrix} c_{1j} \\ c_{2j} \\ c_{3j} \\ c_{4j} \end{bmatrix}, \quad j = 1, 2, 3, 4.$$

Then  $c^{(j)}$  is found by the solution of the equation

$$Ac^{(j)} = I^{(j)}.$$



For example,  $c^{(1)}$  is found by solution of

$$A c^{(1)} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

If operations are to be carried out on the same matrix over a period of time, it might be convenient to record the elements of the matrix on a magnetic card for rapid input at a later date.

#### References:

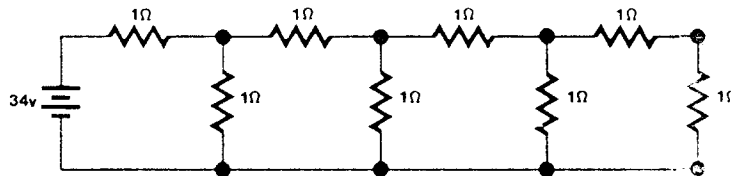
George E. Forsythe, Michael A. Malcolm, and Cleve B. Moler, *Computer Methods in Mathematical Computation*, Computer Science Department, Stanford University, 1972.

G. Forsythe and C. Moler, *Computer Solution of Linear Algebraic Systems*, Prentice-Hall, 1967.

C. Moler, "Matrix Computations with Fortran and Paging," *Comm, ACM*, vol. 15, no. 4, pp. 268-270 (April, 1972).

#### Example 1:

By applying the technique of loop currents to the circuit below, find the currents  $I_1$ ,  $I_2$ ,  $I_3$ , and  $I_4$ .



The equations to be solved are

$$\begin{array}{ccccccc} 2I_1 & -I_2 & & & = & 34 \\ -I_1 & +3I_2 & -I_3 & & = & 0 \\ & -I_2 & +3I_3 & -I_4 & = & 0 \\ & & -I_3 & +3I_4 & = & 0 \end{array}$$

In matrix form,

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 3 & -1 & 0 \\ 0 & -1 & 3 & -1 \\ 0 & 0 & -1 & 3 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 34 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

## Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 029  
 [XEQ] [ALPHA] 4\*4 [ALPHA]  
 2 [R/S]  
 1 [CHS] [R/S]  
 0 [R/S]  
 0 [R/S]  
 1 [CHS] [R/S]  
 3 [R/S]  
 1 [CHS] [R/S]  
 0 [R/S]  
 0 [R/S]  
 1 [CHS] [R/S]  
 3 [R/S]  
 1 [CHS] [R/S]  
 0 [R/S]  
 0 [R/S]  
 1 [CHS] [R/S]  
 3 [R/S]  
 [R/S]  
 [C]  
 34 [R/S]  
 0 [R/S]  
 0 [R/S]  
 0 [R/S]  
 [R/S]  
 [R/S]\*  
 [R/S]\*  
 [R/S]\*

## Display:

a1,1=?  
 a2,1=?  
 a3,1=?  
 a4,1=?  
 a1,2=?  
 a2,2=?  
 a3,2=?  
 a4,2=?  
 a1,3=?  
 a2,3=?  
 a3,3=?  
 a4,3=?  
 a1,4=?  
 a2,4=?  
 a3,4=?  
 a4,4=?  
 READY  
 READY  
 b1,1=?  
 b2,1=?  
 b3,1=?  
 b4,1=?  
 READY  
 X1,1=21.0000 (I<sub>1</sub>)  
 X2,1=8.0000 (I<sub>2</sub>)  
 X3,1=3.0000 (I<sub>3</sub>)  
 X4,1=1.0000 (I<sub>4</sub>)

\*[R/S] is omitted if printer is present.

**Example 2:**

Find the determinant and inverse of the 4x4 matrix in the previous problem.

Continuing from the previous page:

**Keystrokes:**

[D]

[E]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

[R/S]

**Display:**

DET=34.0000

c1,1=0.6176

c2,1=0.2353

c3,1=0.0882

c4,1=0.0294

c1,2=0.2353

c2,2=0.4706

c3,2=0.1765

c4,2=0.0588

c1,3=0.0882

c2,3=0.1765

c3,3=0.4412

c4,3=0.1471

c1,4=0.0294

c2,4=0.0588

c3,4=0.1471

c4,4=0.3824

# User Instructions

				SIZE: 029
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load the program.			
2	Enter matrix A.		[XEQ] 4*4	a1,1=?
		a <sub>1,1</sub>	[R/S]	a2,1=?
		⋮	⋮	⋮
		a <sub>4,4</sub>	[R/S]	READY
3	To review matrix A.		[B]	a1,1=( )
	Corrections may be made by keying in the		[R/S]	a2,1=( )
	correct number when the erroneous element		⋮	⋮
	element is displayed.		[R/S]	READY
4	To print matrix A (if printer exists)		[///] [b]	a1,1=( )
				⋮
				READY
5	Initialize the calculation routines.		[R/S]	READY
	NOTE: This step must be done before			
	you continue!			
	For Simultaneous Equations			
6	Input matrix b.		[C]	b1,1=?
		b <sub>1,1</sub>	[R/S]	b2,1=?
		⋮	⋮	⋮
		b <sub>4,4</sub>	[R/S]	READY
7	To print matrix b.		[///] [c]	b1,1=( )
	NOTE: matrix b may be reviewed by		⋮	⋮
	pressing [C] and then [+ ] when the			READY
	prompt is displayed to view the			
	numerical value of the element.			
8	Calculate x.		[R/S]*	X1,1=( )



# Program Listings

01♦LBL "4*4"	Initialize	51♦LBL 00	
02 SF 21		52 FIX 0	
03 SF 27		53 CF 29	
04♦LBL A		54 "a"	
05 CF 05		55 FS? 06	
06 CF 06	Input A	56 "b"	
07 CF 07		57 FS? 08	Common Input/ Output Routine
08 CF 08		58 "X"	
09 CF 09		59 FS? 09	
10 GTO 01		60 "c"	
11♦LBL C		61 ARCL 27	
12 CF 05	Input b	62 "f,"	
13 SF 06		63 ARCL 26	
14 CF 07		64 "f="	
15 CF 08		65 FC? 05	
16 CF 09		66 "f?"	
17 GTO 01		67 FIX 4	
18♦LBL b		68 SF 29	
19 ADV	Print A	69 RCL IND	
20 SF 05		28	
21 CF 06		70 FS? 05	
22 SF 07		71 ARCL X	
23 CF 08		72 FC? 07	
24 CF 09		73 PROMPT	
25 GTO 01		74 FS? 07	
26♦LBL c		75 AVIEW	
27 ADV		76 STO IND	
28 SF 05	Print b	28	
29 SF 06		77 ISG 28	
30 SF 07		78 CLD	
31 CF 08		79 ISG 27	
32 CF 09		80 GTO 00	
33 GTO 01		81 FS? 07	
34♦LBL B		82 ADV	
35 SF 05	View A	83 FS? 06	
36 CF 06		84 GTO 02	
37 CF 07		85 1.004	
38 CF 08		86 STO 27	
39 CF 09		87 FS? 09	
40♦LBL 01		88 RTN	
41 1.004		89 ISG 26	
42 FC? 09		90 GTO 00	
43 STO 26	Initialize	91♦LBL 02	
44 STO 27	Input Routine	92 "READY"	
45 5		93 PROMPT	
46 FS? 06		94 FS? 06	
47 1		95 GTO 08	
48 FS? 08		96 0	
49 1		97 STO 00	
50 STO 28		98 1	
		99 STO 21	

# Program Listings

100	STO	24		151	X=Y?
101	RCL	05	Pivot Routine	152	GTO 03
102	ABS			153	10
103	STO	22		154	*
104	2			155	XEQ 11
105	RCL	06		156	2
106	XEQ	10		157	XEQ 12
107	3			158	3
108	RCL	07		159	XEQ 12
109	XEQ	10		160	4
110	4			161	XEQ 12
111	RCL	08		162	*LBL 03
112	XEQ	10		163	RCL 10
113	1			164	CHS
114	RCL	21		165	ST/ 11
115	X=Y?			166	ST/ 12
116	GTO	03		167	RCL 11
117	XEQ	11		168	RCL 14
118	1			169	*
119	XEQ	12		170	ST+ 15
120	2			171	RCL 12
121	XEQ	12		172	RCL 14
122	3			173	*
123	XEQ	12		174	ST+ 16
124	4			175	RCL 11
125	XEQ	12		176	RCL 18
126	*LBL	03		177	*
127	RCL	05		178	ST+ 19
128	CHS			179	RCL 12
129	ST/	06		180	RCL 18
130	ST/	07		181	*
131	ST/	08		182	RCL 20
132	9			183	+
133	STO	25		184	STO 20
134	XEQ	13		185	RCL 15
135	XEQ	13		186	ABS
136	XEQ	13		187	RCL 16
137	2			188	ABS
138	STO	21		189	X<=Y?
139	STO	23		190	GTO 03
140	RCL	10		191	RCL 15
141	ABS			192	RCL 16
142	STO	22		193	X<> 15
143	3			194	STO 16
144	RCL	11		195	RCL 19
145	XEQ	10		196	X<> 20
146	4			197	STO 19
147	RCL	12		198	.4
148	XEQ	10		199	XEQ 11
149	2			200	*LBL 03
150	RCL	21		201	RCL 15

# Program Listings

202 CHS		251 RCL IND	
203 ST/ 16		25	
204 RCL 19		252 RTN	
205 RCL 16		253♦LBL 13	
206 *		254 RCL IND	
207 ST+ 20		25	
208 GTO 02		255 STO 21	
209♦LBL 10		256 ISG 25	
210 ABS		257 CLD	
211 RCL 22		258 RCL 06	
212 X>Y?		259 XEQ 09	
213 RTN		260 RCL 07	
214 RDN		261 XEQ 09	
215 STO 22		262 RCL 08	
216 RDN		263♦LBL 09	
217 STO 21		264 RCL 21	
218 RTN		265 *	
219♦LBL 11		266 ST+ IND	
220 ST+ 00		25	
221 RCL 24		267 ISG 25	
222 CHS		268 RTN	
223 STO 24		269 RTN	
224 RTN		270♦LBL D	
225♦LBL 12		271 ADV	
226 STO 22		272 RCL 24	Determinant
227 RCL 23		273 RCL 05	Routine
228 RCL 22		274 *	
229 XEQ 14		275 RCL 10	
230 RCL 21		276 *	
231 RCL 22		277 RCL 15	
232 XEQ 14		278 *	
233 X<>Y		279 RCL 20	
234 STO IND		280 *	
25		281 "DET="	
235 X<>Y		282 ARCL X	
236 RCL 23		283 AVIEW	
237 RCL 22		284 RTN	
238 4		285♦LBL 08	
239 *		286 RCL 00	Simultaneous
240 +		287 10	Equations
241 STO 25		288 STO 23	Routine
242 RDN		289 /	
243 STO IND		290 FRC	
25		291 RCL 23	
244 RTN		292 *	
245♦LBL 14		293 INT	
246 4		294 X=0?	
247 *		295 GTO 03	
248 +		296 STO 25	
249 STO 25		297 RCL IND	
250 CLX		25	



# Program Listings

```

298 X<> 01
299 STO IND
25
300*LBL 03
301 RCL 01
302 RCL 06
303 *
304 ST+ 02
305 RCL 01
306 RCL 07
307 *
308 ST+ 03
309 RCL 01
310 RCL 08
311 *
312 ST+ 04
313 RCL 00
314 RCL 23
315 /
316 INT
317 X=0?
318 GTO 03
319 STO 25
320 RCL IND
25
321 X<> 02
322 STO IND
25
323*LBL 03
324 RCL 12
325 RCL 11
326 RCL 02
327 *
328 ST+ 03
329 CLX
330 RCL 02
331 *
332 ST+ 04
333 RCL 00
334 FRC
335 RCL 23
336 *
337 X=0?
338 GTO 03
339 STO 25
340 RCL IND
25
341 X<> 03
342 STO IND
25
343*LBL 03

```

```

344 RCL 16
345 RCL 03
346 *
347 ST+ 04
348 RCL 20
349 ST/ 04
350 RCL 04
351 CHS
352 STO 21
353 RCL 15
354 STO 22
355 RCL 19
356 RCL 18
357 RCL 17
358 RCL 21
359 *
360 ST+ 01
361 CLX
362 RCL 21
363 *
364 ST+ 02
365 CLX
366 RCL 21
367 *
368 ST+ 03
369 RCL 22
370 ST/ 03
371 RCL 03
372 CHS
373 STO 21
374 RCL 10
375 STO 22
376 RCL 14
377 RCL 13
378 RCL 21
379 *
380 ST+ 01
381 CLX
382 RCL 21
383 *
384 ST+ 02
385 RCL 22
386 ST/ 02
387 RCL 09
388 RCL 02
389 CHS
390 *
391 ST+ 01
392 RCL 05
393 ST/ 01
394 SF 05

```

# Program Listings

```
395 SF 05
396 SF 06
397 FS? 09
398 CF 06
399 SF 07
400 SF 08
401 FC? 09
402 GTO 01
403 XEQ 01
404 ISG 26
405♦LBL 07
406 CLX
407 STO 01
408 STO 02
409 STO 03
410 STO 04
411 RTN
412♦LBL E
413 ADV
414 SF 09
415 XEQ 07
416 1
417 STO 01
418 XEQ 08
419 1
420 STO 02
421 XEQ 08
422 1
423 STO 03
424 XEQ 08
425 1
426 STO 04
427 XEQ 08
428 END
```

Inverse Routine



## 4 X 4 MATRIX OPERATIONS

HEWLETT PACKARD  
SOLUTION BOOK  
HIGH LEVEL MATH

PROGRAM REGISTERS NEEDED: 103

ROW 1 (1 : 4)



ROW 2 (5 : 11)



ROW 3 (11 : 17)



ROW 4 (18 : 24)



ROW 5 (25 : 31)



ROW 6 (32 : 38)



ROW 7 (38 : 43)



ROW 8 (44 : 52)



ROW 9 (52 : 58)



ROW 10 (59 : 64)



ROW 11 (64 : 70)



ROW 12 (70 : 77)



ROW 13 (78 : 85)



ROW 14 (85 : 90)



ROW 15 (91 : 96)



ROW 16 (97 : 105)



ROW 17 (106 : 113)



ROW 18 (113 : 120)



## 4 X 4 MATRIX OPERATIONS

HEWLETT PACKARD  
SOLUTION BOOK  
HIGH LEVEL MATH

ROW 19 (120 : 126)



ROW 20 (127 : 135)



ROW 21 (135 : 140)



ROW 22 (141 : 149)



ROW 23 (149 : 156)



ROW 24 (157 : 163)



ROW 25 (164 : 173)



ROW 26 (174 : 182)



ROW 27 (183 : 191)



ROW 28 (192 : 198)



ROW 29 (199 : 206)



ROW 30 (206 : 215)



ROW 31 (216 : 224)



ROW 32 (224 : 231)



ROW 33 (231 : 238)



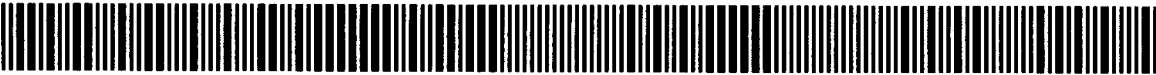
ROW 34 (238 : 248)



ROW 35 (249 : 257)



ROW 36 (257 : 265)



# 4 X 4 MATRIX OPERATIONS

# HEWLETT PACKARD SOLUTION BOOK HIGH LEVEL MATH

ROW 37 (265 : 273)



ROW 38 (274 : 282)



ROW 39 (282 : 291)



ROW 40 (292 : 299)



ROW 41 (300 : 309)



ROW 42 (310 : 319)



ROW 43 (320 : 328)



ROW 44 (329 : 338)



ROW 45 (339 : 345)



ROW 46 (346 : 354)



ROW 47 (355 : 361)



ROW 48 (362 : 370)



ROW 49 (370 : 379)



ROW 50 (379 : 387)



ROW 51 (387 : 396)



ROW 52 (396 : 402)



ROW 53 (403 : 412)



ROW 54 (412 : 419)



## 4 X 4 MATRIX OPERATIONS

HEWLETT PACKARD  
SOLUTION BOOK  
HIGH LEVEL MATH

ROW 55 (420 : 427)



ROW 56 (427 : 428)



## Hewlett-Packard Software

In terms of power and flexibility, the problem-solving potential of the HP-41 programmable calculator is nearly limitless. And in order to see the practical side of this potential, HP has 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 HP-41, HP has an extensive library of "Application Pacs". These programs transform your HP-41 into a specialized calculator in seconds. Included in these pacs are detailed manuals with examples, miniature plug-in Application Modules, and keyboard overlays. Every Application Pac has been designed to extend the capabilities of the HP-41.

You can choose from:

Aviation (Pre-Flight Only) 00041-15018  
Clinical Lab 00041-15024  
Circuit Analysis 00041-15024  
Financial Decisions 00041-15004  
Mathematics 00041-15003  
Structural Analysis 00041-15021  
Surveying 00041-15005  
Securities 00041-15026

Statistics 00041-15002  
Stress Analysis 00041-15027  
Games 00041-15022  
Home Management 00041-15023  
Machine Design 00041-15020  
Navigation 00041-15017  
Real Estate 00041-15016  
Thermal and Transport Science 00041-15019  
Petroleum Fluids 00041-15039

## Users' Library

The Users' Library provides the best programs from contributors and makes them available to you. By subscribing to the HP-41 Users' Library you'll have at your fingertips literally hundreds of different programs from many different application areas.

## \*Users' Library Solutions Books

Hewlett-Packard offers a wide selection of Solutions Books complete with user instructions, examples, and listings. These solution books will complement our other software offerings and provide you with a valuable tool for program solutions.

You can choose from:

Business Stat/Marketing/Sales 00041-90094  
Home Construction Estimating 00041-90096  
Lending, Saving and Leasing 00041-90086  
Real Estate 00041-90136  
Small Business 00041-90137  
Geometry 00041-90084  
High-Level Math 00041-90083  
Test Statistics 00041-90082  
Antennas 00041-90093  
Chemical Engineering 00041-90100  
Control Systems 00041-90092  
Electrical Engineering 00041-90088  
Fluid Dynamics and Hydraulics 00041-90139  
Games II 00041-90443

Civil Engineering 00041-90089  
Heating, Ventilating & Air Conditioning 00041-90140  
Mechanical Engineering 00041-90090  
Solar Engineering 00041-90138  
Calendars 00041-90145  
Cardiac/Pulmonary 00041-90097  
Chemistry 00041-90102  
Games 00041-90099  
Optometry I (General) 00041-90143  
Optometry II (Contact Lens) 00041-90144  
Physics 00041-90142  
Surveying 00041-90141  
Time Module Solutions 00041-90395

\*Some books require additional memory modules to accommodate all programs.



## HIGH-LEVEL MATH

SINE, COSINE, EXPONENTIAL INTEGRALS  
EIGENVALUES/VECTORS OF 3RD - ORDER SYSTEMS  
EIGENVALUES FOR 3RD ORDER SYSTEMS  
CHEBYSHEV, LEGENDRE, HERMITE, AND LAGUERRE POLYNOMIALS  
SIXTEEN-POINT GAUSSIAN QUADRATURE  
GAMMA FUNCTION  
BESSEL FUNCTIONS, ERROR FUNCTION  
CHARACTERISTIC EQUATION OF 4 X 4 MATRIX  
4 X 4 MATRIX OPERATIONS

