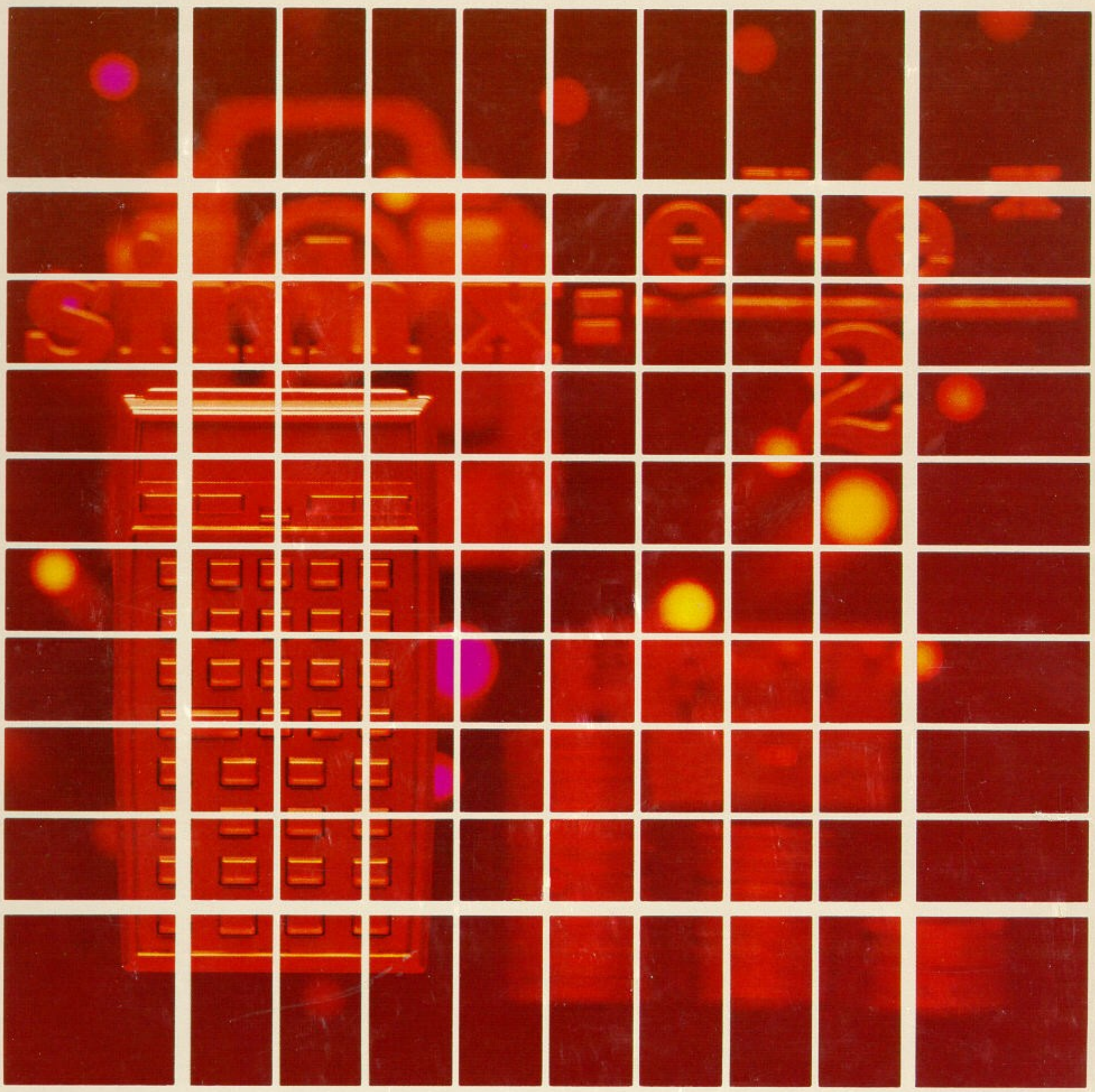
Includes barcode for easy software entry.

HEWLETT-PACKARD

HP-41

USERS' LIBRARY SOLUTIONS

High Level Math



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 ⁎. When you see ⁎ in the program listing, press [×].

   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 fy 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	Keystrokes	Display
01♦LBL "SAMPLE"	■ [LBL] [ALPHA] SAMPLE [ALPHA]	***01 LBL^{T}SAMPLE***
02 "THIS IS A "	[ALPHA] ***THIS IS A*** [ALPHA]	***02^{T}THIS IS A***
03 "├SAMPLE"	[ALPHA] ■ [APPEND] SAMPLE	***03^{T} ├ SAMPLE***
04 AVIEW	■ AVIEW [ALPHA]	***04 AVIEW***
05 6	6	***05 6***
06 ENTER↑	[ENTER↑]	***06 ENTER ↗***
07 -2	2 [CHS]	***07 -2***
08 /	[÷]	***08 /***
09 ABS	[XEQ] [ALPHA] ABS [ALPHA]	***09 ABS***
10 STO IND L	[STO] ■ [•] L	***10 STO IND L***
11 "R3="	[ALPHA] R3= ■ [ARCL] 03	***11^{T}R3=***
12 ARCL 03	■ [AVIEW]	***12 ARCL 03***
13 AVIEW	[ALPHA]	***13 AVIEW***
14 RTN	■ [RTN]	***14 RTN***

TABLE OF CONTENTS

* 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.

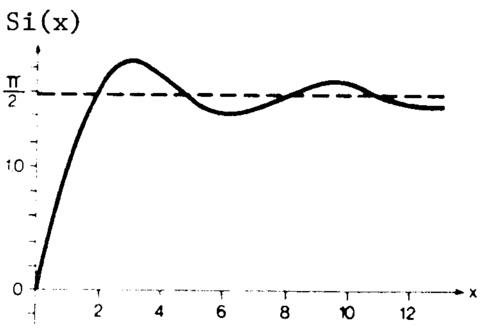
Exponential Integral

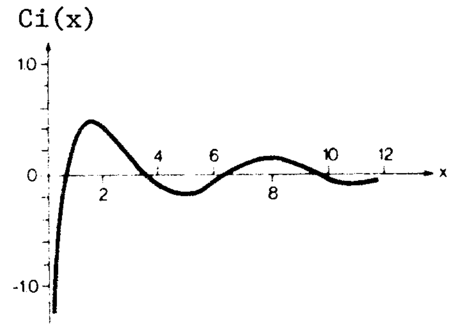
$$Ei(x) = \int_{-\infty}^{x} \frac{e^t}{t} \, dt = \gamma + \ln x + \sum_{n=1}^{\infty} \frac{x^n}{n \; n!}$$

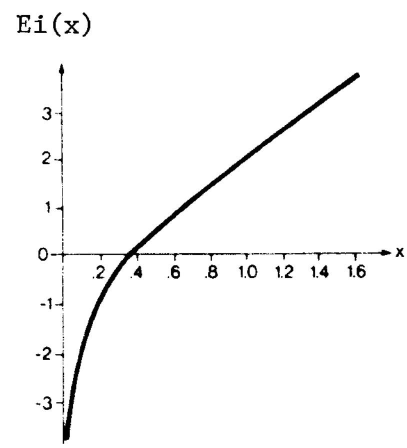
where $x > 0$, and $\gamma = 0.5772156649$ is Euler's constant.

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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 004	
[XEQ] [ALPHA] SI [ALPHA]	
1. .69 [A]	SI(0.69)=0.67
2. 9.8 [A]	SI(9.80)=1.67
3. 1.38 [B]	CI(1.38)=0.46
4. 5 [B]	CI(5.00)=-0.19
5. 1.59 [C]	EI(1.59)=3.57
6. .61 [C]	EI(0.61)=0.80

User Instructions

SIZE: 004

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Key in the program.			
2.	Initialize.		[XEQ] SI	
3.	To calculate:			
	•Si(x)	x	[A]	SI(x)=( )
	•Ci(x)	x	[B]	CI(x)=( )
	•Ei(x)	x	[C]	EI(x)=( )

Program Listings

Initialize

```
01♦LBL "SI"
02 SF 27
03 STOP
```

Si(x)

```
04♦LBL A
05 STO 02
06 STO 03
07 X↑2
08 CHS
09 STO 00
10 1
11 STO 01
12 RCL 02
13 "S"
```

Loop to add terms

```
14♦LBL 00
15 RCL 00
16 RCL 01
17 1
18 +
19 /
20 LASTX
21 XEQ 02
22 X≠Y?
23 GTO 00
24 GTO 04
```

Common subroutine

```
25♦LBL 02
26 1
27 +
28 STO 01
29 /
30 RCL 02
31 *
32 STO 02
33 RCL 01
34 /
35 +
36 RTN
```

Ci(x)

```
37♦LBL B
38 STO 03
39 X↑2
40 CHS
41 STO 00
42 1
43 STO 02
44 0
45 STO 01
46 LASTX
47 XEQ 01
48 "C"
49 GTO 00
50♦LBL 01
51 LN
52 .5772156
649
53 +
54 RTN
```

Ei(x)

```
55♦LBL C
56 STO 03
57 STO 00
58 1
59 STO 02
60 0
61 STO 01
62 RCL 00
63 XEQ 01
64♦LBL 03
65 RCL 00
66 RCL 01
67 XEQ 02
68 X≠Y?
69 GTO 03
70 "E"
```

Display routine

```
71♦LBL 04
72 FIX 2
73 "⊢I<"
74 ARCL 03
75 "⊢>="
76 ARCL X
77 AVIEW
78 END
```

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	$-x^2$, x	50	
01	temp. storage		
02	temp. storage		
03	x		
05		55	
10		60	
15		65	
20		70	
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 004 TOT. REG. 22 USER MODE

ENG ___ FIX 2 SCI ___ ON X OFF ___

DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
27		User On	User Off

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Si(x)	A		
Ci(x)	B		
Ei(x)	C		

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 $Ax = \lambda x$

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

(a) Power method

Assume the eigenvalues of A are λ_1, λ_2 and λ_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 $v = c_1v_1 + c_2v_2 + c_3v_3$ then

$$Av = c_1Av_1 + c_2Av_2 + c_3Av_3 = \lambda_1 \left(c_1v_1 + c_2 \frac{\lambda_2}{\lambda_1} v_2 + c_3\frac{\lambda_3}{\lambda_1} v_3\right)$$

$$A^pv = \lambda_1^p \left[c_1v_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^p v_2 + c_3 \left(\frac{\lambda_3}{\lambda_1}\right)^p v_3\right]$$

therefore $$\lambda_1 = \lim_{p\to\infty} \frac{(A^{p+1}v_1)}{(A^pv_1)}$$

(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:

Given $A = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix}$ and $v_1$guess $= \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$, find λ_1, v_1, λ_2, v_2, λ_3, and v_3.

Keystrokes:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 028	
[///] [FIX] 4	
[XEQ] [ALPHA] EVV [ALPHA]	a1,1=?
3 [CHS] [R/S]	a2,1=?
2 [R/S]	a3,1=?
0 [R/S]	a1,2=?
1 [R/S]	a2,2=?
3 [CHS] [R/S]	a3,2=?
1 [R/S]	a1,3=?
0 [R/S]	a2,3=?
2 [R/S]	a3,3=?
3 [CHS] [R/S]	READY
[C]	V1,1=?
1 [R/S]	V1,2=?
1 [R/S]	V1,3=?
1 [R/S]	READY
[R/S]	LAM.1=-5.0000
[R/S]*	V1,1=1.0000
[R/S]*	V1,2=-2.0000
[R/S]*	V1,3=1.0000
[R/S]*	LAM.2=-3.0000
[R/S]*	V2,1=1.0000
[R/S]*	V2,2=-1.0000E-9
[R/S]*	V2,3=-1.0000
[R/S]*	LAM.3=-1.0000
[R/S]*	V3,1=1.0000
[R/S]*	V3,2=2.0000
[R/S]*	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=?
		$a_{1,1}$	[R/S]	a2,1=?
		$a_{2,1}$	[R/S]	a3,1=?
		⋮	⋮	⋮
		$a_{2,3}$	[R/S]	a3,3=?
		$a_{3,3}$	[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_1. This step must be performed		[C]	V1,1=?
	regardless of whether or not v_1 is	$v_{1,1}$	[R/S]	V1,2=?
	different from that of the previous	$v_{1,2}$	[R/S]	V1,3=?
	calculation.	$v_{1,3}$	[R/S]	READY
6	Print v_1.		[///] [C]	V1,1=( )
				⋮
				READY
7	Calculate v_1, v_2, v_3, λ_1, λ_2, λ_3. 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=( )

User Instructions

				SIZE:
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
			[R/S]*	V2,1=( )
			[R/S]*	V2,2=( )
			[R/S]*	V2,3=( )
			[R/S]*	LAM.3=( )
			[R/S]*	V3,1=( )
			[R/S]*	V3,2=( )
			[R/S]*	V3,3=( )
8	Reinput matrix A. Note that the value		[A]	a1,1=?
	of each element from the previous matrix	$a_{1,1}$	[R/S]	a2,1=?
	A may be reused by just pressing [R/S].	⋮	⋮	⋮
		$a_{3,3}$	[R/S]	READY
	Steps 2, 3, 5 and 8 may be interrupted			
	at any point to execute steps 3, 4, 5, 6			
	or 8.			
	*[R/S] is omitted if printer is present.			

Program Listings

Program	Comment
01♦LBL "EVV" 02 SF 21 03 SF 27	
04♦LBL A 05 CF 05 06 CF 06 07 CF 07 08 GTO 01	Input A matrix
09♦LBL b 10 ADV 11 SF 05 12 CF 06 13 SF 07 14 GTO 01	Print A matrix
15♦LBL C 16 CF 05 17 SF 06 18 CF 07 19 GTO 01	Input v_1
20♦LBL c 21 ADV 22 SF 05 23 SF 06 24 SF 07 25 GTO 01	Print v_1
26♦LBL B 27 SF 05 28 CF 06 29 CF 07	View A matrix
30♦LBL 01 31 1.003 32 STO 10 33 STO 11 34 1 35 FS? 06 36 20 37 STO 00	Initialize Input/View routine
38♦LBL 00 39 FIX 0 40 CF 29 41 "a" 42 FS? 06 43 "V1," 44 ARCL 11 45 FC? 06 46 "⊢," 47 FC? 06 48 ARCL 10 49 "⊢=" 50 FC? 05	Common Input/View routine
51 "⊢?" 52 FIX 4 53 SF 29 54 RCL IND 00 55 FS? 05 56 ARCL X 57 FC? 07 58 PROMPT 59 FS? 07 60 AVIEW 61 STO IND 00 62 ISG 00 63 CLD 64 ISG 11 65 GTO 00 66 FS? 07 67 ADV 68 FS? 06 69 GTO 02 70 1.003 71 STO 11 72 ISG 10 73 GTO 00	
74♦LBL 02 75 "READY" 76 PROMPT 77 "A" 78 ASTO 26	"READY" prompt
79♦LBL 03 80 RCL 01 81 RCL 04 82 RCL 07 83 XEQ 10 84 STO 23 85 RCL 02 86 RCL 05 87 RCL 08 88 XEQ 10 89 STO 24 90 RCL 03 91 RCL 06 92 RCL 09 93 XEQ 10 94 STO 25 95 RCL 23 96 RCL 20 97 / 98 RCL 26 99 X=Y?	Iterative routine to find λ_1

Program Listings

```
100 GTO 04
101 X<>Y
102 STO 26
103 RCL 23
104 STO 20
105 RCL 24
106 STO 21
107 RCL 25
108 STO 22
109 GTO 03
```

Interation subroutine

```
110♦LBL 10
111 RCL 22
112 *
113 RCL 21
114 ST* Z
115 RDN
116 RCL 20
117 ST* T
118 RDN
119 +
120 +
121 RTN
```

Calculate v_1

```
122♦LBL 04
123 CF 09
124 1
125 STO 00
126 STO 27
127 STO 20
128 RCL 24
129 RCL 23
130 /
131 STO 21
132 RCL 25
133 RCL 23
134 /
135 STO 22
136 RCL 26
137 RDN
138 XEQ 06
```

Calculate λ_2

```
139 RCL 05
140 RCL 21
141 RCL 04
142 *
143 -
144 STO 16
145 RCL 06
146 RCL 22
147 RCL 04
148 *
149 -
150 STO 17
151 RCL 08
152 RCL 21
153 RCL 07
154 *
155 -
156 STO 18
157 RCL 09
158 RCL 22
159 RCL 07
160 *
161 -
162 STO 19
163 RCL 16
164 +
165 STO 25
166 X↑2
167 RCL 16
168 RCL 19
169 *
170 RCL 18
171 RCL 17
172 *
173 -
174 4
175 *
176 -
177 SQRT
178 2
179 /
180 STO 23
181 RCL 25
182 2
183 /
184 STO 24
185 RCL 23
186 -
187 STO 10
188 RCL 24
189 RCL 23
190 +
191 STO 13
```

Common routine to calculate v_2 and v_3

```
192♦LBL 08
193 RCL 10
194 RCL 16
195 -
196 STO 12
197 RCL 18
198 STO 11
199 X<>Y
200 RCL 04
201 RCL 07
```

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
```

Output routine

```
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
```

Output subroutine

```
268♦LBL 07
269 FIX 0
270 CF 29
271 ARCL 00
272 FC? 08
273 GTO 07
274 "⊢,"
275 ARCL 27
276 ISG 27
277♦LBL 07
278 FIX 4
279 SF 29
280 "⊢="
281 .END.
```

80	
90	
00	

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	Index/Temp.	50	
	$a_{1,1}$		
	$a_{2,1}$		
	$a_{3,1}$		
	$a_{1,2}$		
05	$a_{2,2}$	55	
	$a_{3,2}$		
	$a_{1,3}$		
	$a_{2,3}$		
	$a_{3,3}$		
10	Index/Temp.	60	
	Index/Temp.		
	Temp.		
	Temp.		
	Temp.		
15	Temp.	65	
	Temp.		
	Temp.		
	Temp.		
	Temp.		
20	Temp.	70	
	Temp.		
	Temp.		
	Temp.		
	Temp.		
25	Temp.	75	
	Temp.		
	Temp.		
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 28 TOT. REG. 97 USER MODE
ENG ___ FIX 4 SCI ___ ON X OFF ___
DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
05			
06			
07			
08			
21		Printer Enabled	Printer Disabled
27		User Mode On	User Mode Off
29		Digit Grouping	No Digit Grouping

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Input A	A	Print A	b
View A	B	Print V_1	c
Input V_1	C		

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)

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)

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 λ_1, λ_2, and λ_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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 022	
[XEQ] [ALPHA] EV [ALPHA]	a1,1=?
1 [R/S]	a2,1=?
0 [R/S]	a3,1=?
0 [R/S]	a1,2=?
0 [R/S]	a2,2=?
0 [R/S]	a3,2=?
1 [R/S]	a1,3=?
0 [R/S]	a2,3=?
1 [CHS] [R/S]	a3,3=?
.5 [CHS] [R/S]	READY
[C]	RLAM.1=-0.2500
[R/S]*	ILAM.1=-0.9682
[R/S]*	RLAM.2=-0.2500

Keystrokes:	Display:
[R/S]*	ILAM.2=0.9682
[R/S]*	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=?
		$a_{1,1}$	[R/S]	a2,1=?
		$a_{2,1}$	[R/S]	a3,1=?
		⋮	⋮	⋮
		$a_{3,1}$	[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 λ.		[R/S]*	ILAM.1=( )
	ILAM. ≡ the imaginary portion of λ.		[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

Listing	Comment
01◆LBL "EV"	
02 SF 21	
03 SF 27	
04◆LBL A	
05 CF 05	Input A
06 CF 07	
07 GTO 01	
08◆LBL b	
09 ADV	
10 SF 05	Print A
11 SF 07	
12 GTO 01	
13◆LBL B	
14 SF 05	View A
15 CF 07	
16◆LBL 01	
17 1.003	
18 STO 10	Initialize Input Routine
19 STO 11	
20 1	
21 STO 00	
22◆LBL 00	
23 FIX 0	
24 CF 29	
25 "a"	
26 ARCL 11	Input/View/ Print Routine
27 "⊢,"	
28 ARCL 10	
29 "⊢="	
30 FC? 05	
31 "⊢?"	
32 FIX 4	
33 SF 29	
34 RCL IND 00	
35 FS? 05	
36 ARCL X	
37 FC? 07	
38 PROMPT	
39 FS? 07	
40 AVIEW	
41 STO IND 00	
42 ISG 00	
43 CLD	
44 ISG 11	
45 GTO 00	
46 FS? 07	
47 ADV	
48 1.003	
49 STO 11	

Listing	Comment
50 ISG 10	
51 GTO 00	
52◆LBL 02	
53 "READY"	
54 PROMPT	
55◆LBL C	
56 RCL 08	
57 RCL 06	
58 RCL 05	
59 RCL 09	Calculate coefficients of 3rd order polynomial
60 XEQ 09	
61 RCL 01	
62 *	
63 STO 10	
64 RCL 02	
65 RCL 09	
66 RCL 08	
67 RCL 03	
68 XEQ 09	
69 RCL 04	
70 *	
71 ST+ 10	
72 RCL 03	
73 RCL 05	
74 RCL 06	
75 RCL 02	
76 XEQ 09	
77 RCL 07	
78 *	
79 ST+ 10	
80 RCL 01	
81 RCL 05	
82 RCL 07	
83 RCL 03	
84 XEQ 09	
85 STO 11	
86 RCL 01	
87 RCL 09	
88 RCL 08	
89 RCL 06	
90 XEQ 09	
91 ST+ 11	
92 RCL 05	
93 RCL 09	
94 RCL 04	
95 RCL 02	
96 XEQ 09	
97 ST+ 11	
98 RCL 01	
99 RCL 05	
100 RCL 09	

Program Listings

```
101 +
102 +
103 CHS
104 STO 12
105 CF 06
106 RCL 10
107 X≠0?
108 GTO 10
109 STO 21
```

Test if constant is **zero**

```
110 RCL 11
111 STO 17
112 RCL 12
113 STO 18
114 GTO 07
115♦LBL 10
116 RCL 11
117 3
118 *
119 RCL 12
120 X↑2
121 -
122 9
123 /
124 STO 13
125 RCL 11
126 RCL 12
127 *
128 9
```

Solve for Real Root λ_3

```
129 *
130 RCL 10
131 27
132 *
133 -
134 RCL 12
135 3
136 Y↑X
137 ST+ X
138 -
139 54
140 /
141 STO 14
142 X↑2
143 RCL 13
144 3
145 Y↑X
146 +
147 X<0?
148 GTO 03
149 SQRT
150 STO 19
151 RCL 14
152 +
153 XEQ 05
```

S+T

```
154 RCL 14
155 RCL 19
156 -
157 XEQ 05
158 +
159 GTO 04
160♦LBL 03
161 RCL 14
162 RCL 13
163 3
164 Y↑X
165 CHS
```

λ_3 for $Q^3+R^2<0$

```
166 SQRT
167 /
168 ACOS
169 3
170 /
171 COS
172 RCL 13
173 CHS
174 SQRT
175 *
176 ST+ X
177♦LBL 04
178 RCL 12
179 3
180 /
```

λ_3

```
181 -
182 STO 19
183 STO 21
184 RCL 12
185 +
186 STO 18
187 RCL 10
188 RCL 19
```

Reduce to second order and calculate λ_1, λ_2

```
189 /
190 CHS
191 STO 17
192♦LBL 07
193 RCL 18
194 X↑2
195 RCL 17
196 4
197 *
198 -
199 CHS
200 X>0?
201 SF 06
202 ABS
203 SQRT
```

Program Listings

```
204 2
205 /
206 STO 20
207 RCL 18
208 CHS
209 2
210 /
211 STO 00
212 FS? 06
213 GTO 11
214 RCL 20
215 +
216 0
217 X<>Y
218 RCL 00
219 RCL 20
220 -
221 0
222 X<>Y
223 GTO 06
224♦LBL 11
225 RCL 20
226 X<>Y
227 RCL 20
228 CHS
229 RCL 00
230 GTO 06
231♦LBL 09
232 *
```

ZT-XY

```
233 RDN
234 *
235 R↑
236 -
237 RTN
238♦LBL 05
239 CF 08
240 X<0?
```

$\pm\sqrt[3]{\ }$

```
241 SF 08
242 ABS
243 3
244 1/X
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
255 "I"
256 SF 05
257 XEQ 08
258 "R"
259 SF 07
260 XEQ 08
261 "I"
262 SF 07
263 XEQ 08
264 CLA
265 SF 09
266 RCL 21
267♦LBL 08
268 "⊢LAM."
269 FS?C 05
270 "⊢1"
271 FS?C 07
272 "⊢2"
273 FS?C 09
274 "⊢3"
275 "⊢="
276 ARCL X
277 AVIEW
278 RDN
279 END
```

80

90

00

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	-6 (or b b/2)	50	
	a_{11}		
	a_{21}		
	a_{31}		
	a_{12}		
05	a_{22}	55	
	a_{32}		
	a_{13}		
	a_{23}		
	a_{33}		
10	λ_3	60	
	used		
	used		
	used		
	used		
15	used	65	
	used		
	used		
	used		
	used		
20	used	70	
	λ_3		
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 22 TOT. REG. 86 USER MODE

ENG ____ FIX 4 SCI ____ ON X OFF ____

DEG ____ RAD ____ GRAD ____

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
05		Display Elements	Input Elements
07		Print Elements	Any other mode.

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Input A	A	Print A	b
View A	B	Compute λ's	C

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)

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

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

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

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

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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 007	
[XEQ] [ALPHA] T [ALPHA]	N?
3 [R/S]	X?
.4 [R/S]	T3(0.40)=-0.94
[XEQ] [ALPHA] P [ALPHA]	N?
10 [R/S]	X?
0.98 [R/S]	P10(0.98)=0.16
[XEQ] [ALPHA] H [ALPHA]	N?
5 [R/S]	X?
3 [R/S]	H5(3.00)=3,816.00
[XEQ] [ALPHA] L [ALPHA]	N?
6 [R/S]	X?
3 [R/S]	L6(3.00)=-0.01
■ [FIX] 4	-0.0125

User Instructions

	SIZE: 007

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load program			
2	For Chebyshev polynomial		[XEQ] T	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	T(n)(x)=
3	For Legendre polynomial		[XEQ] P	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	P(n)(x)=
4	For Hermite polynomial		[XEQ] H	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	H(n)(x)=
5	For Laguerre polynomial		[XEQ] L	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	L(n)(x)=

Program Listings

Chebyshev polynomial

```
◆LBL "T"
XEQ A
STO 00
2
STO 01
*
STO 02
CLX
X<>Y
X<=Y?
GTO 00
1
STO 03
X=Y?
GTO 01
```

Loop for recurrence equation

```
◆LBL 02
CLX
RCL 02
RCL 00
*
RCL 03
LASTX
STO 03
RDN
-
STO 00
CLX
RCL 01
1
+
STO 01
```

Test, $r_1 \leq n_1$ so back to loop

```
X<=Y?
GTO 02
◆LBL 01
RCL 00
"T"
GTO d
◆LBL 00
1
"T"
GTO d
```

Legendre polynomial

```
◆LBL "P"
XEQ A
STO 01
STO 04
CLX
X<>Y
STO 00
X<=Y?
GTO 00
1
STO 03
X<>Y
X<=Y?
GTO 04
2
STO 02
```

Loop for recurrence equation

```
◆LBL 03
RCL 04
RCL 01
*
ENTER↑
ENTER↑
RCL 03
-
+
LASTX
RCL 04
STO 03
CLX
RCL 02
/
-
STO 04
RCL 00
RCL 02
1
+
STO 02
```

Test

```
X<=Y?
GTO 03
RCL 04
"P"
GTO d
◆LBL 04
RCL 01
RTN
```

Hermite polynomial

```
◆LBL H
XEQ A
STO 01
2
*
STO 03
CLX
X<>Y
X<=Y?
GTO 00
1
STO 00
STO 02
X<>Y
X<=Y?
```

Program Listings

```
103 GTO 05
```

Test for recurrence equation

```
104♦LBL 06
105 RCL 00
106 RCL 03
107 STO 00
108 RCL 01
109 *
110 X<>Y
111 RCL 02
112 *
113 -
114 2
115 *
116 STO 03
117 CLX
118 RCL 02
119 1
120 +
121 STO 02
122 X<>Y
```

Test

```
123 X≠Y?
124 GTO 06
125 RCL 03
126 "H"
127 GTO d
128♦LBL 05
129 RCL 03
130 RTN
```

Laguerre polynomial

```
131♦LBL "L"
132 XEQ A
133 ENTER↑
134 1
135 STO 03
136 +
137 STO 01
138 2
139 STO 02
140 X<>Y
141 -
142 STO 04
143 CLX
144 X<>Y
145 X<=Y?
146 GTO 00
147 STO 00
148 1
149 X<>Y
150 X<=Y?
151 GTO 07
```

Loop for recurrence equation

```
152♦LBL 08
153 RCL 04
154 RCL 03
155 -
156 RCL 04
157 +
158 RCL 01
159 RCL 04
160 *
161 RCL 03
162 -
163 RCL 02
164 /
165 RCL 04
166 STO 03
167 RDN
168 -
169 STO 04
170 RCL 00
171 RCL 02
172 1
173 +
174 STO 02
```

Test

```
175 X<=Y?
176 GTO 08
177♦LBL 07
178 RCL 04
179 "L"
180 GTO d
181 STOP
```

Input n and x

```
182♦LBL A
183 "N?"
184 PROMPT
185 STO 05
186 "X?"
187 PROMPT
188 STO 06
189 RTN
```

Display

```
190♦LBL d
191 CF 29
192 FIX 0
193 ARCL 05
194 "├<"
195 SF 29
196 FIX 2
197 ARCL 06
198 "├>="
199 ARCL X
200 AVIEW
201 STOP
202 .END.
```

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	temp. change	50	
	temp. change		
	temp. change		
	temp. change		
05	n	55	
	x		
10		60	
15		65	
20		70	
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 007 TOT. REG. 45 USER MODE

ENG ___ FIX 2 SCI ___ ON ___ OFF X

DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
		NONE	

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
NONE			

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

$\int_a^b f(x)\,dx$ or $\int_a^\infty f(x)\,dx$ 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 <u>plus</u> 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:	Display:
[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]	01 LBL ᵀX
[1/x]	02 1/X
[///] [x_2]	03 X/2
[///] [GTO] ..	
[PRGM]	
[XEQ] [ALPHA] GAUSS [ALPHA]	
[A]	NAME?
X [R/S]	a?
1 [R/S]	b?
4 [R/S]	0.7500
[B]	NAME?
X [R/S]	a?
1 [R/S]	1.0000

User Instructions

SIZE: 021

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load the program.			
2	Load the data (registers 1-16).			
3	Key in the desired function.		[///][GTO] . .	
			⋮	
			[///][GTO] . .	
4	Initialize "GAUSS".		[XEQ] GAUSS	
5	For a finite interval (a→b)		[A]	NAME?
	Input name of program from step 3.	Name	[R/S]	a?
	Input lower bound.	a	[R/S]	b?
	Input upper bound.	b	[R/S]	$\int_a^b$ (Name)
6	For an infinite interval (a→∞)		[B]	NAME?
	Input name of program from step 3.	Name	[R/S]	a?
	Input lower bound.	a	[R/S]	$\int_a^b$ (Name)
7	To evaluate the function at a given point.	x	[XEQ] "Name"	f(x)
8	For a new function go to step 3.			
9	For new bounds go to step 5 or 6.			

Program Listings

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

Program Listings

```
 96 DSE 19
 97 RCL IND
19
 98 X<>Y
 99 /
100 *
101 ST+ 00
102 END
```

10	
20	
30	
40	
50	
51	
60	
70	
80	
90	
00	

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	Sum	50	
	W_{15}, W_{16}		
	Z_{15}, $-Z_{16}$		
	W_{13}, W_{14}		
	Z_{13}, $-Z_{14}$		
05	W_{11}, W_{12}	55	
	Z_{11}, $-Z_{12}$		
	W_9, W_{10}		
	Z_9, $-Z_{10}$		
	W_7, W_8		
10	Z_7, $-Z_8$	60	
	W_5, W_6		
	Z_5, $-Z_6$		
	W_3, W_4		
	Z_3, $-Z_4$		
15	W_1, W_2	65	
	Z_1, $-Z_2$		
	b		
	a		
	Index		
20	Function Name	70	
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 21 TOT. REG. 46 USER MODE
ENG ____ FIX ____ SCI ____ ON X OFF ____
DEG ____ RAD ____ GRAD ____

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
05		Integral from a to b	Integral from a to ∞
06		Negative loop	Positive loop

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Integral from a to b	A	Integral from a to ∞	B

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)

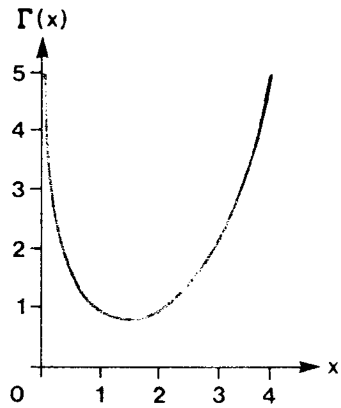
ROW 12 (85 : 94)

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.

   $\Gamma(x) \simeq 1 + b_1 (x - 1) + b_2 (x - 1)^2 + \ldots + b_8 (x - 1)^8$

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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 001	
[///] [FIX] 2	
5.25 [XEQ] [ALPHA] GAMMA [ALPHA]	GAMMA=35.21
8 [R/S]	GAMMA=5,040.00
3.34 [R/S]	GAMMA=2.80

User Instructions

SIZE: 001

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load the program.			
2	Input x.	x	[XEQ] GAMMA	GAMMA=( )
3	For new x.	x	[R/S]	GAMMA=( )

Program Listings

```
01♦LBL "GAMMA"
02 1
03 -
04 "ILLEGAL X"
05 X<0?                (x-1)<0, error
06 GTO 01
07 INT
08 LASTX
09 X=Y?
10 GTO 02
11 1
12 STO 00
13 X<>Y
14♦LBL 03
15 X<=Y?               (x-1)(x-2)
16 GTO 00              (x-3) ...
17 ST* 00              until <1
18 1
19 -
20 GTO 03
21♦LBL 00
22 ENTER↑
23 ENTER↑
24 ENTER↑
25 .035868343          polynomial
26 *                   approximation
27 .193527818
28 -
29 *
30 .482199394
31 +
32 *
33 .756704078
34 -
35 *
36 .918206857
37 +
38 *
39 .897056937
40 -
41 *
42 .988205891
43 +
44 *
45 .577191652
46 -
47 *
48 1
49 +
50 RCL 00
51 *
52 GTO 04
53♦LBL 02
54 FACT
55♦LBL 04
56 "GAMMA="            Output
57 ARCL X
58♦LBL 01
59 AVIEW
60 .END.
```

70

80

90

00

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	Γ	50	
05		55	
10		60	
15		65	
20		70	
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 001 TOT. REG. 29 USER MODE

ENG ____ FIX ____ SCI ____ ON ____ OFF ____

DEG ____ RAD ____ GRAD ____

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY

GAMMA FUNCTION

PROGRAM REGISTERS NEEDED: 24

HEWLETT PACKARD
SOLUTION BOOK:
HIGH ORDER MATH

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 $x > \emptyset$. 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

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

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

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

After calculating a $J_n(x)$, the values of $J_o(x)$ 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

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

and the complementary error function as

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

For large values of x (≥3), the error function is very close to 1. If erfc(x) is computed as 1 - erf(x), most of the significant figures of 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 ≤ 3, the error function is computed by a series sum

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

and the complementary error function

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

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

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

and the error function by

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

The accuracy of the calculation of erf(x) and 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 $0 \le x \le 10^{-6}$ 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 erfc(x) will halt on overflow for $x \geqslant 15$.

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 erf and 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	

User Instructions

SIZE: 007

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load program and set USER mode		[USER]	
2	Initialize		[XEQ] INIT	
3	To find $J_n(x)$		[J]	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	J=
	(optional) to find $J_0(x)$		[R/S]	J0=
	and $J_1(x)$		[R/S]	J1=
4	To find $I_n(x)$		[I]	N?
	input n	n	[R/S]	X?
	input x	x	[R/S]	I=
5	To find error		[E]	ACCURACY?
	input no. of decimal places	n	[R/S]	X?
	input x	x	[R/S]	ERF=
			[R/S]	ERFC=

Program Listings

```
 01◆LBL "INI          "J"
T"
 02 CLRG
 03 RTN
 04◆LBL J
 05 XEQ a
 06 SF 09
 07◆LBL 09
 08 XEQ b
 09 CF 02
 10 ST+ 00
 11 XEQ b
 12 FS?C 02
 13 GTO 09
 14 RCL 03
 15 RCL 00
 16 ENTER↑
 17 +
 18 RCL 05
 19 -
 20 /
 21 "J="
 22 XEQ d               - - - - - - - - -
 23 GTO C               input n + x
 24◆LBL a
 25 FIX 4
 26 "N?"
 27 PROMPT
 28 STO 01
 29 "X?"
 30 PROMPT              - - - - - - - - -
 31 1.5                 initialization
 32 *                   for Bessel
 33 STO 03              (Jn + In)
 34 RCL 01
 35 X<=Y?
 36 X<>Y
 37 6
 38 +
 39 RCL 03
 40 9
 41 *
 42 RCL 03
 43 2
 44 +
 45 /
 46 +
 47 2
 48 /
 49 INT
 50 ENTER↑
```

```
 51 +
 52 2
 53 +
 54 STO 06
 55 3
 56 RCL 03
 57 /
 58 STO 02
 59 0
 60 STO 05
 61 STO 00
 62  E-9
 63 STO 04
 64 RTN                 - - - - - - - - -
 65◆LBL b               Compute one term
 66 DSE 06              FZ set except
 67 SF 02               for F = 0
 68 RCL 06
 69 RCL 01
 70 X≠Y?
 71 GTO 00
 72 RCL 04
 73 STO 03
 74◆LBL 00
 75 RDN
 76 RCL 05
 77 FS? 00
 78 CHS
 79 X<>Y
 80 RCL 02
 81 *
 82 RCL 04
 83 STO 05
 84 *
 85 +
 86 STO 04
 87 RTN                 - - - - - - - - -
 88◆LBL C               Compute J0(x) +
 89 RCL 05              J1(x)
 90 RCL 00
 91 ENTER↑
 92 +
 93 RCL 05
 94 -
 95 /
 96 "J0="
 97 XEQ d
 98 RCL 04
 99 CHS
100 RCL 00
101 ENTER↑
102 +
```

Program Listings

```
103 RCL 05
104 -
105 /
106 "J1="
107 GTO d
108♦LBL I            "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            Compute error
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
```

```
152 X>Y?
153 GTO 03
154♦LBL 07           Loop for erf
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           Exit erf
174 LASTX
175 RCL 04
176 /
177 2
178 *
179 1
180 X<>Y
181 -
182 LASTX
183 GTO 02
184♦LBL 03           Find erfc
185 RCL 02           x > 3
186 1/X
187 STO 02
188 RCL 01
189 1/X
190 STO 01
191♦LBL 06           Loop for erfc
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
```

Program Listings

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

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	ΣT_k	50	
	n; erf term		
	2/x		
	1.5x, Tn		
	T_k; $(e^{x^2}\sqrt{\pi})^{-1}$		
05	T_{k+1}	55	
	K; places		
10		60	
15		65	
20		70	
25		75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 007 TOT. REG. 53 USER MODE
ENG ___ FIX 09 SCI ___ ON X OFF ___
DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY

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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 026	
[XEQ] [ALPHA] CEM [ALPHA]	$a_{1,1}$=?
1 [R/S]	$a_{1,2}$=?
0 [R/S]	$a_{1,3}$=?
1 [R/S]	$a_{1,4}$=?
0 [R/S]	$a_{2,1}$=?
1 [R/S]	$a_{2,2}$=?
0 [R/S]	$a_{2,3}$=?
2 [R/S]	$a_{2,4}$=?
1 [CHS] [R/S]	$a_{3,1}$=?
3 [R/S]	$a_{3,2}$=?
1 [CHS] [R/S]	$a_{3,3}$=?
0 [R/S]	$a_{3,4}$=?
2 [R/S]	$a_{4,1}$=?
2 [CHS] [R/S]	$a_{4,2}$=?
1 [CHS] [R/S]	$a_{4,3}$=?
1 [CHS] [R/S]	$a_{4,4}$=?
0 [R/S]	R1=-1.000
[R/S]*	R2=0.000
[R/S]*	R3=7.000
[R/S]*	R4=2.000

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

User Instructions

SIZE: 026

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1	Load program.			
2	Begin execution.		[XEQ] CEM	$a_{1,1}$=?
	Input the matrix.	$a_{1,1}$	[R/S]	$a_{1,2}$=?
		$a_{1,2}$	[R/S]	$a_{1,3}$=?
		⋮	⋮	⋮
		$a_{4,3}$	[R/S]	$a_{4,4}$=?
		$a_{4,4}$	[R/S]	R1=( )
			[R/S]*	R2=( )
			[R/S]*	R3=( )
			[R/S]*	R4=( )
	*[R/S] is omitted if printer is attached.			

Program Listings

```
01♦LBL "CEM
"
```

Initialize

```
02 SF 21
03 SF 27
04♦LBL A
```

Input matrix

```
05 CF 05
06 CF 07
07 GTO 03
08♦LBL b
```

Print matrix

```
09 ADV
10 SF 05
11 SF 07
12 GTO 03
13♦LBL B
```

View matrix

```
14 SF 05
15 CF 07
16♦LBL 03
```

Initialize
Input Routine

```
17 1.004
18 STO 01
19 STO 02
20 10
21 STO 00
22♦LBL 04
23 FIX 0
24 CF 29
```

Common Input
Routine

```
25 "a"
26 ARCL 01
27 "⊦,"
28 ARCL 02
29 "⊦="
30 FC? 05
31 "⊦?"
32 FIX 4
33 SF 29
34 RCL IND
00
35 FS? 05
36 ARCL X
37 FC? 07
38 PROMPT
39 FS? 07
40 AVIEW
41 STO IND
00
42 ISG 00
43 CLD
44 ISG 02
45 GTO 04
46 FS? 07
47 ADV
48 1.004
49 STO 02
50 ISG 01
51 GTO 04
52 "READY"
53 PROMPT
54♦LBL C
```

Calculation of r_1

```
55 RCL 10
56 STO 00
57 RCL 15
58 +
59 RCL 20
60 +
61 RCL 25
62 +
63 CHS
64 "R1"
65 XEQ 02
```

Calculation of r_2

```
66 RCL 10
67 RCL 20
68 +
69 RCL 15
70 *
71 RCL 10
72 RCL 25
73 +
74 RCL 20
75 *
76 +
77 RCL 10
78 RCL 15
79 +
80 RCL 25
81 *
82 +
83 RCL 17
84 RCL 23
85 *
86 -
87 RCL 21
88 RCL 24
89 *
90 -
91 RCL 16
92 RCL 19
93 *
94 -
95 RCL 11
96 RCL 14
97 *
98 -
99 RCL 12
```

Program Listings

```
100 RCL 18
101 *
102 -
103 RCL 13
104 RCL 22
105 *
106 -
107 "R2"
```

Calculate r_3

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

Program Listings

```
202 "R3"
203 XEQ 02
204 RCL 14
205 STO 01
206 RCL 18
207 STO 04
208 RCL 22
209 STO 07
210 XEQ 00
211 RCL 11
212 *
213 ST- 00
214 RCL 15
215 STO 02
216 RCL 19
217 STO 05
218 RCL 23
219 STO 08
220 XEQ 00
221 RCL 12
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 *
233 ST- 00
234 RCL 00
235 "R4"
236♦LBL 02
237 "⊢="
238 ARCL X
239 AVIEW
240 RTN
241♦LBL 00
242 RCL 05
243 RCL 09
244 *
245 RCL 06
246 RCL 08
247 *
248 -
249 RCL 01
250 *
251 RCL 06
252 RCL 07
```

Calculate r_4

Output routine

Find

$$\begin{vmatrix} R_1 & R_2 & R_3 \\ R_4 & R_5 & R_6 \\ R_7 & R_8 & R_9 \end{vmatrix}$$

```
253 *
254 RCL 04
255 RCL 09
256 *
257 -
258 RCL 02
259 *
260 +
261 RCL 04
262 RCL 08
263 *
264 RCL 05
265 RCL 07
266 *
267 -
268 RCL 03
269 *
270 +
271 .END.
```

70

80

90

00

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	Temp. storage	50	
	Temp.		
	Temp.		
	Temp.		
	Temp.		
05	Temp.	55	
	Temp.		
	Temp.		
	Temp.		
	Temp.		
10	$a_{1,1}$	60	
	$a_{1,2}$		
	$a_{1,3}$		
	$a_{1,4}$		
	$a_{2,1}$		
15	$a_{2,2}$	65	
	$a_{2,3}$		
	$a_{2,4}$		
	$a_{3,1}$		
	$a_{3,2}$		
20	$a_{3,3}$	70	
	$a_{3,4}$		
	$a_{4,1}$		
	$a_{4,2}$		
	$a_{4,3}$		
25	$a_{4,4}$	75	
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 026 TOT. REG. 86 USER MODE
ENG ___ FIX 4 SCI ___ ON X OFF ___
DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
05		View Matrix	Input Matrix
07		Print Matrix	
21		Printer Enabled	Printer Disabled
27		User Mode Set	User Mode Clear

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Input Matrix	A	Print Matrix	b
View Matrix	B	Calculate	c

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)

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 ($i \le j$) 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 A x = 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 4x4 matrix such that AC = CA = I, where I is the 4x4 identity matrix.

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

let $c^{(j)}$ be the j^{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:

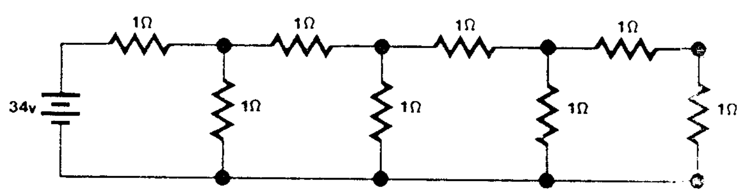
George E. Forsythe, Michael A. Malcolm, and Cleve B. Moler, Computer Methods in Mathematical Computation, Computer Science Department, Stanford University, 1972.

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

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

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:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 029	
[XEQ] [ALPHA] 4*4 [ALPHA]	a1,1=?
2 [R/S]	a2,1=?
1 [CHS] [R/S]	a3,1=?
0 [R/S]	a4,1=?
0 [R/S]	a1,2=?
1 [CHS] [R/S]	a2,2=?
3 [R/S]	a3,2=?
1 [CHS] [R/S]	a4,2=?
0 [R/S]	a1,3=?
0 [R/S]	a2,3=?
1 [CHS] [R/S]	a3,3=?
3 [R/S]	a4,3=?
1 [CHS] [R/S]	a1,4=?
0 [R/S]	a2,4=?
0 [R/S]	a3,4=?
1 [CHS] [R/S]	a4,4=?
3 [R/S]	READY
[R/S]	READY
[C]	b1,1=?
34 [R/S]	b2,1=?
0 [R/S]	b3,1=?
0 [R/S]	b4,1=?
0 [R/S]	READY
[R/S]	X1,1=21.0000 (I_1)
[R/S]*	X2,1=8.0000 (I_2)
[R/S]*	X3,1=3.0000 (I_3)
[R/S]*	X4,1=1.0000 (I_4)

*[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:	Display:
[D]	DET=34.0000
[E]	c1,1=0.6176
[R/S]	c2,1=0.2353
[R/S]	c3,1=0.0882
[R/S]	c4,1=0.0294
[R/S]	c1,2=0.2353
[R/S]	c2,2=0.4706
[R/S]	c3,2=0.1765
[R/S]	c4,2=0.0588
[R/S]	c1,3=0.0882
[R/S]	c2,3=0.1765
[R/S]	c3,3=0.4412
[R/S]	c4,3=0.1471
[R/S]	c1,4=0.0294
[R/S]	c2,4=0.0588
[R/S]	c3,4=0.1471
[R/S]	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=?
		$a_{1,1}$	[R/S]	a2,1=?
		⋮	⋮	⋮
		$a_{4,4}$	[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=?
		$b_{1,1}$	[R/S]	b2,1=?
		⋮	⋮	⋮
		$b_{4,4}$	[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=( )

User Instructions

				SIZE:
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
			⋮	⋮
			[R/S]*	READY
	For Determinant			
9	To calculate Det A		[D]	DET=( )
	For Inverse A			
10	To calculate C.		[E]	c1,1=( )
			[R/S]*	c2,1=( )
			⋮	⋮
			[R/S]*	c4,1=( )
11	For new problem.		[A]	a1,1=?
	NOTE: This is the same as step 2.		⋮	⋮
	Go to step 3.			
	*[R/S] is omitted if printer is present.			

Program Listings

Initialize

```
 01♦LBL "4*4
"
 02 SF 21
 03 SF 27
```

Input A

```
 04♦LBL A
 05 CF 05
 06 CF 06
 07 CF 07
 08 CF 08
 09 CF 09
 10 GTO 01
```

Input b

```
 11♦LBL C
 12 CF 05
 13 SF 06
 14 CF 07
 15 CF 08
 16 CF 09
 17 GTO 01
```

Print A

```
 18♦LBL b
 19 ADV
 20 SF 05
 21 CF 06
 22 SF 07
 23 CF 08
 24 CF 09
 25 GTO 01
```

Print b

```
 26♦LBL c
 27 ADV
 28 SF 05
 29 SF 06
 30 SF 07
 31 CF 08
 32 CF 09
 33 GTO 01
```

View A

```
 34♦LBL B
 35 SF 05
 36 CF 06
 37 CF 07
 38 CF 08
 39 CF 09
```

Initialize Input Routine

```
 40♦LBL 01
 41 1.004
 42 FC? 09
 43 STO 26
 44 STO 27
 45 5
 46 FS? 06
 47 1
 48 FS? 08
 49 1
 50 STO 28
```

Common Input/ Output Routine

```
 51♦LBL 00
 52 FIX 0
 53 CF 29
 54 "a"
 55 FS? 06
 56 "b"
 57 FS? 08
 58 "X"
 59 FS? 09
 60 "c"
 61 ARCL 27
 62 "⊦,"
 63 ARCL 26
 64 "⊦="
 65 FC? 05
 66 "⊦?"
 67 FIX 4
 68 SF 29
 69 RCL IND
28
 70 FS? 05
 71 ARCL X
 72 FC? 07
 73 PROMPT
 74 FS? 07
 75 AVIEW
 76 STO IND
28
 77 ISG 28
 78 CLD
 79 ISG 27
 80 GTO 00
 81 FS? 07
 82 ADV
 83 FS? 06
 84 GTO 02
 85 1.004
 86 STO 27
 87 FS? 09
 88 RTN
 89 ISG 26
 90 GTO 00
 91♦LBL 02
 92 "READY"
 93 PROMPT
 94 FS? 06
 95 GTO 08
 96 0
 97 STO 00
 98 1
 99 STO 21
```

Program Listings

Pivot Routine

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

Program Listings

```
202 CHS
203 ST/ 16
204 RCL 19
205 RCL 16
206 *
207 ST+ 20
208 GTO 02
209♦LBL 10
210 ABS
211 RCL 22
212 X>Y?
213 RTN
214 RDN
215 STO 22
216 RDN
217 STO 21
218 RTN
219♦LBL 11
220 ST+ 00
221 RCL 24
222 CHS
223 STO 24
224 RTN
225♦LBL 12
226 STO 22
227 RCL 23
228 RCL 22
229 XEQ 14
230 RCL 21
231 RCL 22
232 XEQ 14
233 X<>Y
234 STO IND
25
235 X<>Y
236 RCL 23
237 RCL 22
238 4
239 *
240 +
241 STO 25
242 RDN
243 STO IND
25
244 RTN
245♦LBL 14
246 4
247 *
248 +
249 STO 25
250 CLX
```

```
251 RCL IND
25
252 RTN
253♦LBL 13
254 RCL IND
25
255 STO 21
256 ISG 25
257 CLD
258 RCL 06
259 XEQ 09
260 RCL 07
261 XEQ 09
262 RCL 08
263♦LBL 09
264 RCL 21
265 *
266 ST+ IND
25
267 ISG 25
268 RTN
269 RTN
```

Determinant Routine

```
270♦LBL D
271 ADV
272 RCL 24
273 RCL 05
274 *
275 RCL 10
276 *
277 RCL 15
278 *
279 RCL 20
280 *
281 "DET="
282 ARCL X
283 AVIEW
284 RTN
```

Simultaneous Equations Routine

```
285♦LBL 08
286 RCL 00
287 10
288 STO 23
289 /
290 FRC
291 RCL 23
292 *
293 INT
294 X=0?
295 GTO 03
296 STO 25
297 RCL IND
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          Inverse Routine
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
```

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

00	Pivots	50	
	b1		
	b2		
	b3		
	b4		
05	a1	55	
	a2, m21		
	a3, m31		
	a4, m41		
	a5		
10	a6	60	
	a7, m32		
	a8, m42		
	a9		
	a10		
15	a11	65	
	a12, m43		
	a13		
	a14		
	a15		
20	a16	70	
	temp storage		
	temp storage		
	10		
	±1		
25	temp storage	75	
	Index		
	Index		
	Index		
30		80	
35		85	
40		90	
45		95	

STATUS

SIZE 029 TOT. REG. 132 USER MODE
ENG ___ FIX 4 SCI ___ ON X OFF ___
DEG ___ RAD ___ GRAD ___

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
05		View matrix	Input matrix
06		Print matrix	
07		Matrix b	
08		Output c	
09		Output I	
21		Printer Enabled	Printer Disabled
27		User On	User Off

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY
Input A	A	Print A	b
View A	B	Print b	c
Input b	C	Inv.	E
Det.	D		

4 X 4 MATRIX OPERATIONS

PROGRAM REGISTERS NEEDED: 103

HEWLETT PACKARD
SOLUTION BOOK
HIGH LEVEL MATH

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)

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)

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 accomodate 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

HEWLETT PACKARD

00041-90083 Rev. D 11/82 Printed in U.S.A.