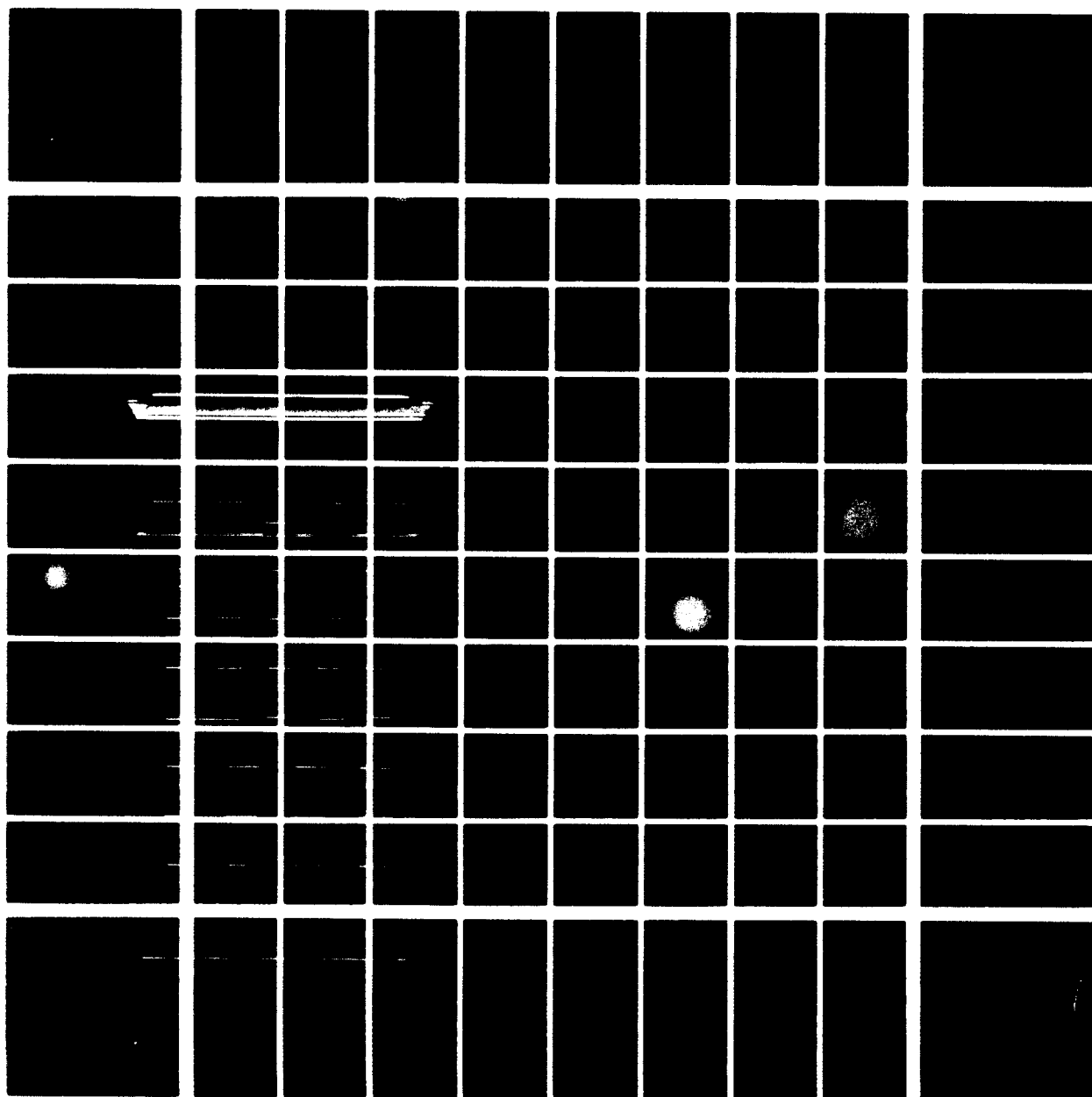


Includes barcode for easy software entry.

HEWLETT-PACKARD

HP-41

USERS' LIBRARY SOLUTIONS
Calendars



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.

00041-90145 Rev. E 5/83 Printed in U.S.A.

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 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	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	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

1.	CALENDAR DATE/JULIAN DATE CONVERSIONS	1
	Converts between Gregorian and Julian calendars.	
2.	DAY OF YEAR - DAY OF WEEK	9
	Calculates the day of year and day of week.	
3.	NUMBER OF WEEKDAYS BETWEEN TWO DATES	17
	Calculates the number of weekdays between any two dates in history.	
4.	IN WHAT YEAR IS A GIVEN DATE AN M-DAY?	24
	User specifies a year, and a date, then a day-of-week. Program calculates what years from the specified year have the specified date falling on a specific day.	
5.	NUMBER OF M-DAYS BETWEEN TWO DATES, AND THE N-TH M-DAY OF THE MONTH	29
	User specifies a day-of-week and two dates; the program calculates the number of times that day occurs between the given dates. User specifies a year, a month, a day-of-week, and a number "N"; the program then calculates the N-th occurrence of the specified day in the given month.	
6.	HOLIDAYS	37
	Calculates the date, in a given year, for Mother's Day, Father's Day, Election Day, Thanksgiving Day, Washington's Birthday, Labor Day, Columbus Day, and Veteran's Day.	
7.	RELIGIOUS HOLIDAYS	45
	Calculates the date, in a given year, for Ash Wednesday, First Sunday in Lent, Passion Sunday, Palm Sunday, Good Friday, Easter, Rogation Sunday, Ascension Day, Whitsunday, and Trinity Sunday.	
8.	CHINESE YEARS TO/FROM GREGORIAN YEARS	53
	Converts the cyclical two-character year designation of the Chinese Calendar to/from Gregorian years.	
*9.	NEW MOON AND FULL MOON DAY OF MONTH	59
	Calculates the dates of new and full moons.	
**10.	CALENDAR PRINTOUT	68
	Calculates and prints a calendar for a given year and the number of months.	

*These programs require an additional memory module

**This program requires a printer and two memory modules.

CALENDAR DATE TO JULIAN DATE CONVERSION

This program converts any given calendar date between March 1, 1900 and February 28, 2100, to a Julian Date (a continuous count of days from an epoch in the very distant past). For example, January 1, 1979, yield a Julian Date of 2,443,875. Thus the interval between any two calendar dates (on the Gregorian calendar) can be found by obtaining the Julian Date for each and then finding the absolute value of the difference between the two.

To calculate the Julian Day number, the following equation is used:

$$\text{Julian Day number} = \text{INT} (365.25 y') + \text{INT} (30.6001 m') + d + 1,720,982$$

where

$$y' = \begin{cases} \text{year} - 1 & \text{if } m = 1 \text{ or } 2 \\ \text{year} & \text{if } m > 2 \end{cases}$$

$$m' = \begin{cases} \text{month} + 13 & \text{if } m = 1 \text{ or } 2 \\ \text{month} + 1 & \text{if } m > 2 \end{cases}$$

This program will also convert any Julian Date (greater than 0) to its corresponding Gregorian Date.

NOTE: If any discrepancy is noticed, see the reference for further details.

REFERENCE: O'Neil, W. M., Time and the Calendars, Sydney Univ. Press, Australia, 1975.

Example 1:

Determine the Julian Day Number for July 4, 1979.

Keystrokes:

Display:

[XEQ] [ALPHA] SIZE [ALPHA] 007

[XEQ] [ALPHA] JUL [ALPHA]

MM.DDYyyy ?

7.041979 [R/S]

JD=2,444,059.

Example 2:

Given the Julian Date 2,444,233, what is the corresponding Gregorian Date?

Keystrokes:

Display:

[XEQ] [ALPHA] CAL [ALPHA]

JULIAN DAY?

2444233 [R/S]

12-25-1979

Keystrokes:

[XEQ] [ALPHA] JUL [ALPHA]

8.281940 [R/S]

[STO] 06

[XEQ] [ALPHA] JUL [ALPHA]

8.281978 [R/S]

[RCL] 06

[-]

Display:

MM.DDYYYY ?

JD=2,429,870.

2,429,870.0000

MM.DDYYYY ?

JD=2,443,749.

2,429,870.0000

13,879.0000

User Instructions

[illegible]

Program Listings

01♦LBL "JUL "	Initialization and prompting for input	48 FIX 0	Initialization and prompting for input
02 FIX 0		49 "JULIAN DAY?"	
03 "MM.DDYY YY ?"		50 PROMPT	
04 PROMPT		51 CF 29	
05 INT		52 ENTER↑	
06 STO 00	Unpack month,day, and year. Then store into appro- priate register.	53 68569	
07 LASTX		54 +	L=Julian Date + 68569
08 FRC		55 STO 04	
09 1 E2		56 4	
10 *		57 *	N=4*L/146097
11 INT		58 146097	
12 STO 01		59 /	
13 LASTX		60 INT	
14 FRC		61 STO 05	
15 1 E4		62 146097	
16 *		63 *	
17 STO 02		64 3	
18 2		65 +	
19 RCL 00	Calculate m ,and y'	66 4	
20 X>Y?		67 /	
21 GTO 00		68 INT	
22 1		69 CHS	
23 ST- 02		70 RCL 04	
24 12		71 +	
25 ST+ 00		72 STO 04	
26♦LBL 00		73 1	
27 1		74 +	
28 ST+ 00		75 4000	
29 RCL 02		76 *	
30 365.25		77 1461001	I=4000*(L+1)/ 1461001
31 *		78 /	
32 INT	Calculate day number	79 INT	
33 30.6001		80 STO 03	
34 RCL 00		81 1461	L=L-1461*I/4+31
35 *		82 *	
36 INT		83 4	
37 +		84 /	
38 RCL 01		85 INT	
39 +		86 CHS	
40 1720982		87 31	
41 +		88 +	
42 "JD="	Display result	89 RCL 04	
43 ARCL X		90 +	
44 AVIEW		91 STO 04	J=80*L/2447
45 FIX 4		92 80	
46 STOP		93 *	
47♦LBL "CAL "		94 2447	
		95 /	
		96 INT	
		97 STO 02	

Program Listings

98	2447		51	
99	*			
100	80			
101	/			
102	INT	K=L-2447*J/80		
103	CHS			
104	RCL 04			
105	+			
106	STO 01			
107	RCL 02			
108	11		60	
109	/	L=J/11		
110	INT			
111	STO 04			
112	12			
113	*			
114	CHS			
115	2	J=J+2-12*L		
116	+			
117	RCL 02			
118	+		70	
119	STO 02			
120	RCL 05			
121	49	I=100*(N-49)+I+L		
122	-			
123	1 E2			
124	*			
125	RCL 03			
126	+			
127	RCL 04			
128	+		80	
129	STO 03			
130	CLA			
131	ARCL 02	Display result		
132	"F--"	in proper format		
133	ARCL 01			
134	"F--"			
135	ARCL 03			
136	AVIEW			
137	SF 29			
138	CLX		90	
139	FIX 4			
140	STOP			
141	.END.			
50			00	

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS				STATUS			
00	month	50		SIZE <u>007</u>	TOT. REG. <u>44</u>	USER MODE	
	day			ENG <u> </u>	FIX <u> </u>	SCI <u> </u>	ON <u> </u> OFF <u>X</u>
	year			DEG <u> </u>	RAD <u> </u>	GRAD <u> </u>	
	Term 1			FLAGS # INIT S/C SET INDICATES CLEAR INDICATES			
	Term 2						
05	Term 3	55		29		Radix Point	No Radix Point
	temporary for user						
10		60					
15		65					
20		70					
25		75					
30		80					
35		85					
				ASSIGNMENTS			
				FUNCTION	KEY	FUNCTION	KEY
40		90					
45		95					

CALENDAR DATE/JULIAN DATE
CONVERSIONS
PROGRAM REGISTERS NEEDED: 38

ROW 1 (1 - 3)



ROW 2 (3 - 8)



ROW 3 (9 - 17)



ROW 4 (18 - 26)



ROW 5 (27 - 33)



ROW 6 (33 - 40)



ROW 7 (40 - 43)



ROW 8 (44 - 48)



ROW 9 (49 - 50)



ROW 10 (51 - 58)



ROW 11 (58 - 62)



ROW 12 (62 - 74)



ROW 13 (75 - 78)



ROW 14 (79 - 87)



ROW 15 (88 - 96)



ROW 16 (97 - 105)



ROW 17 (106 - 116)



ROW 18 (117 - 126)



CALENDAR DATE/JULIAN DATE
CONVERSIONS

ROW 19 (127 - 134)



ROW 20 (134 - 141)



DAY OF YEAR - DAY OF WEEK

Given a date (between years 1901 and 2009), or given a year and day of the year, this program will calculate the date, the day of the year, the remaining days of the year, and the day of the week. The day of the year and date are calculated using the following Julian Day formula:

Julian Day number = $\text{INT}(365.25 \ y') + \text{INT}(30.6001 \ m') + d + 1,720,982$

where $y' = \begin{cases} \text{year} - 1 & \text{if } m = 1 \text{ or } 2 \\ \text{year} & \text{if } m > 2 \end{cases}$

$m' = \begin{cases} \text{month} + 13 & \text{if } m = 1 \text{ or } 2 \\ \text{month} + 1 & \text{if } m > 2 \end{cases}$

Example 1:

For July 4, 1979 find:

- 1) Day of the year
- 2) Remaining days in the year
- 3) The day of the week

Keystrokes:

Display:

[XEQ] [ALPHA] SIZE [ALPHA] 011

1) [XEQ] [ALPHA] DOY [ALPHA]

MM.DDYYYY ?

7.041979 [R/S]

185 DAY

2) [XEQ] [ALPHA] RDOY [ALPHA]

180 DAYS

3) [XEQ] [ALPHA] DOW [ALPHA]

WED

Example 2:

For the year 1956, the 315th day, determine the following:

- 1) The date
- 2) The remaining days in the year
- 3) The day of the week

Keystrokes:

Display:

1) [XEQ] [ALPHA] DAT [ALPHA]

YEAR ?

1956 [R/S]

DOY ?

315 [R/S]

11-10-1956

2) [XEQ] [ALPHA] RDOY [ALPHA]

51 DAYS

3) [XEQ] [ALPHA] DOW [ALPHA]

SAT

[illegible]

Program Listings

01♦LBL "DOY"		50 INT	
"		51 RCL 09	
02 XEQ 08		52 RCL 10	
03 "MM.DDYY		53 *	
YY ?"		54 INT	
04 PROMPT	Prompt for date	55 +	
05 XEQ 00		56 RCL 08	
06 STO 02	Compute day	57 +	
07 RCL 04	number	58 RTN	
08 1 E6	Compute day	59♦LBL 04	Make corrections
09 /	number for first	60 INT	to m' and y'
10 1	of year	61 ST+ 09	
11 +		62 12	
12 XEQ 00		63 *	
13 RCL 02		64 -	
14 -	Find the differ-	65 RTN	
15 CHS	ence and display	66♦LBL "DAT	
16 STO 00	result	"	
17 CF 29		67 XEQ 08	
18 CLA		68 "YEAR ?"	
19 ARCL X		69 PROMPT	Prompt for year,
20 SF 29		70 "DOY ?"	and day of year.
21 "F DAY"		71 PROMPT	
22 AVIEW		72 STO 00	
23 RTN		73 RDN	
24♦LBL 00		74 STO 04	
25 ENTER↑	Break date into	75 1 E6	Compute first
26 INT	month, day and	76 /	day of year.
27 STO 07	year.	77 1	
28 -		78 +	
29 1 E2		79 XEQ 00	
30 *		80 RCL 00	
31 ENTER↑		81 +	
32 INT		82 STO 02	Decipher day
33 STO 08		83 RCL 05	number into
34 -		84 -	month day and
35 1 E4		85 RCL 10	year
36 *		86 /	
37 STO 09		87 INT	
38 STO 04		88 STO 09	
39 RCL 07		89 RCL 10	
40 1	Calculate Julian	90 *	
41 +	Day number	91 INT	
42 ENTER↑		92 RCL 02	
43 1/X		93 -	
44 .7		94 CHS	
45 +		95 STO 06	
46 CHS		96 RCL 01	
47 XEQ 04		97 /	
48 RCL 01		98 INT	
49 *		99 STO 07	

Program Listings

100 RCL 06		150 "F DAYS"	
101 X<>Y		151 AVIEW	
102 RCL 01		152 SF 29	
103 *		153 RTN	
104 INT		154♦LBL "DOW	
105 -		"	
106 STO 08		155 RCL 02	
107 RCL 07		156 5	Calculate day of
108 1		157 +	week number
109 RCL 08		158 7	
110 %		159 /	
111 -		160 FRC	
112 -		161 7	
113 RCL 07		162 *	
114 14		163 RND	
115 /		164 X=0?	
116 XEQ 04		165 "SUN"	
117 CF 29		166 1	Display day of
118 CLA		167 X=Y?	week
119 INT		168 "MON"	
120 ARCL X		169 CLX	
121 "F--"		170 2	
122 LASTX		171 X=Y?	
123 FRC	Put date into	172 "TUE"	
124 1 E2	MM-DD-YYYY	173 CLX	
125 *		174 3	
126 ARCL X		175 X=Y?	
127 "F--"		176 "WED"	
128 ARCL 09		177 CLX	
129 AVIEW		178 4	
130 SF 29		179 X=Y?	
131 CLX		180 "THUR"	
132 RTN		181 CLX	
133♦LBL "RDO		182 5	
Y"		183 X=Y?	
134 RCL 04		184 "FRI"	
135 4		185 CLX	
136 /	Compute remain-	186 6	
137 FRC	ing days of the	187 X=Y?	
138 X=0?	year	188 "SAT"	
139 SF 07		189 AVIEW	
140 RCL 03		190 RTN	
141 .5		191♦LBL 08	
142 FS?C 07		192 FIX 0	
143 CHS		193 365.25	Initialize
144 -		194 STO 10	
145 RCL 00		195 .25	
146 -		196 +	
147 CF 29		197 STO 03	
148 CLA		198 30.6001	
149 ARCL X		199 STO 01	

[illegible]

DAY OF YEAR - DAY OF WEEK

PROGRAM REGISTERS NEEDED: 50

ROW 1 (1 - 3)



ROW 2 (3 - 5)



ROW 3 (6 - 14)



ROW 4 (15 - 21)



ROW 5 (21 - 30)



ROW 6 (31 - 41)



ROW 7 (42 - 51)



ROW 8 (52 - 62)



ROW 9 (63 - 67)



ROW 10 (68 - 70)



ROW 11 (70 - 79)



ROW 12 (79 - 91)



ROW 13 (92 - 104)



ROW 14 (105 - 116)



ROW 15 (116 - 123)



ROW 16 (124 - 130)



ROW 17 (130 - 135)



ROW 18 (136 - 145)



DAY OF YEAR - DAY OF WEEK

ROW 19 (146 - 150)



ROW 20 (151 - 156)



ROW 21 (157 - 166)



ROW 22 (167 - 173)



ROW 23 (174 - 180)



ROW 24 (180 - 188)



ROW 25 (188 - 193)



ROW 26 (194 - 198)



ROW 27 (199 - 204)



NUMBER OF WEEKDAYS BETWEEN TWO DATES

This program calculates the number of weekdays between any two dates in history. The program uses dates encoded in MM.DDYYYY form. The number of weekdays is calculated using the following formula:

$$W(m,d,y) = 5 \left\lfloor \frac{D(m,d,y)}{7} \right\rfloor + 1/2 \left\lfloor 1.801 (D(m,d,y) \bmod 7) \right\rfloor$$

$$\text{where: } D(m,d,y) = d - \left\lfloor 3/4 \left(\left\lfloor \frac{g(y,m)}{100} \right\rfloor - 7 \right) \right\rfloor + \left\lfloor 365.25 g(y,m) \right\rfloor + \left\lfloor 30.6 f(m) \right\rfloor$$

$$f(m) = \begin{cases} m + 13 & \text{if } m=1,2 \\ m + 1 & \text{if } m>2 \end{cases} \quad g(y,m) = \begin{cases} y-1 & \text{if } m=1,2 \\ y & \text{if } m>2 \end{cases} \quad \left\lfloor \right\rfloor = \text{INT}$$

The program then calculates the difference between the current W and the previous W. These calculations are from noon-to-noon, so the difference between a weekday and a non-weekday will have a half day in it.

This program is valid from the beginning of the Gregorian Calendar, through the year 9999.

Example 1:

Determine how many weekdays there are between September 2, 1963 and June 5, 1964 and the day of the week that June 5, 1964 falls on.

Keystrokes:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 004	
[XEQ] [ALPHA] WEEK [ALPHA]	DATE 1 ?
9.021963 [R/S]	DATE 2 ?
6.051964 [R/S]	199.0 WEEKDAYS
[R/S]*	DATE 2 = FRI.

Example 2:

Determine how many weekdays between April 1, 1974 and May 1, 1974. Also, what is the day-of-week of the last date entered, May 1, 1974?

Keystrokes:	Display:
[XEQ] [ALPHA] WEEK [ALPHA]	DATE 1 ?
4.011974 [R/S]	DATE 2 ?
5.011974 [R/S]	22.0 WEEKDAYS
[R/S]*	DATE 2 = WED.

*This [R/S] is not necessary if the printer is attached.

[illegible]

Program Listings

01♦LBL "WEE K"		48 X=Y?	
02♦LBL 05		49 "FSAT."	
03 SF 21		50 RDN	
04 FIX 6		51 0	
05 "DATE 1?"	Prompt for 1st date	52 X=Y?	
"		53 "FSUN."	
06 ADV		54 TONE 8	
07 XEQ 02		55 AVIEW	
08 XEQ 00		56 GTO 05	
09 "DATE 2?"	Prompt for 2nd date	57♦LBL 02	
"		58 PROMPT	
10 TONE 8		59 ASTO 00	
11 XEQ 02		60 CLA	
12 XEQ 00		61 ARCL 00	
13 FIX 1		62 "F = "	
14 CLA		63 ARCL X	
15 ARCL X	Displaying output	64 FS? 55	
16 "F WEEKD AYS"		65 PRA	
17 TONE 8		66 RTN	
18 AVIEW		67♦LBL 00	
19 CLA		68 9	
20 ARCL 00		69 ENTER↑	
21 "F = "		70 1 E-6	
22 RCL 03		71 -	
23 7	Determine and display day of week	72 X<>Y	
24 *		73 3	
25 RND		74 X<>Y	
26 INT		75 X<=Y?	M-1 or 2? Add 12 to M & Y = Y-1
27 1		76 XEQ 01	
28 X=Y?		77 1	Add 1 to M
29 "FMON."		78 +	
30 RDN		79 ENTER↑	
31 2		80 INT	Store adjusted M
32 X=Y?		81 STO 01	
33 "FTUE."		82 -	
34 RDN		83 100	
35 3		84 *	
36 X=Y?		85 ENTER↑	.YYYY
37 "FWED."		86 FRC	
38 RDN		87 100	
39 4		88 *	YY.YY
40 X=Y?		89 ENTER↑	C=# of centuries
41 "FTHUR."		90 INT	
42 RDN		91 7	
43 5		92 -	
44 X=Y?		93 .75	
45 "FFRI."		94 *	
46 RDN		95 INT	
47 6		96 X<>Y	
		97 36525	
		98 *	Century days

[illegible]

DATA REGISTERS				STATUS					
00	Alpha prompt f(m)	50		SIZE	004	TOT. REG.	40	USER MODE	
	Code # prev. date			ENG		FIX	1	SCI	
	1/2 day-of-week			DEG		RAD		GRAD	
				FLAGS					
05		55		#	INIT S/C	SET INDICATES	CLEAR INDICATES		
				29		Radix point	No Radix point		
10		60							
15		65							
20		70							
25		75							
30		80							
35		85							
				ASSIGNMENTS					
40		90		FUNCTION		KEY	FUNCTION		KEY
45		95							

NUMBER OF WEEKDAYS BETWEEN
TWO DATES

PROGRAM REGISTERS NEEDED: 37

ROW 1 (1 : 4)



ROW 2 (5 : 8)



ROW 3 (8 : 11)



ROW 4 (11 : 16)



ROW 5 (16 : 20)



ROW 6 (20 : 28)



ROW 7 (29 : 33)



ROW 8 (33 : 39)



ROW 9 (40 : 45)



ROW 10 (45 : 49)



ROW 11 (50 : 56)



ROW 12 (56 : 62)



ROW 13 (63 : 70)



ROW 14 (70 : 80)



ROW 15 (81 : 89)



ROW 16 (90 : 97)



ROW 17 (97 : 106)



ROW 18 (107 : 115)



NUMBER OF WEEKDAYS BETWEEN
TWO DATES

ROW 19 (116 : 127)



ROW 20 (128 : 132)



IN WHAT YEAR IS A GIVEN DATE AN M-DAY?

Given a month m , a day-of-month d , a day-of-week w , (coded 0=SUN.,..., 6=SAT.), and a starting year Y , the program applies the following formula:

$$f(m) = \begin{cases} m+11 & \text{if } m > 2 \\ m+23 & \text{if } m = 1 \text{ or } 2 \end{cases} \quad g(m) = \begin{cases} 1 & \text{if } m=1 \text{ or } 2 \\ 0 & \text{if } m>2 \end{cases}$$

$$N(m,d,w) = (|2.6f(m)| + d - w) \bmod 7$$

$$h(N) = \begin{cases} 4\left(\left\lfloor \frac{N}{2} \right\rfloor + 4\right) & \text{if } N \text{ odd} \\ 2N & \text{if } N \text{ even} \end{cases} \quad \left\lfloor \frac{N}{2} \right\rfloor = \text{INT}$$

$$L(y,N) = (y-1-h(N)) \bmod 28 + h(N) + g(m)$$

L is a leap year, preceding the specified year, in which the given date is an M -day. By successively adding 6,11,6,5,6,11,6,5,... one obtains all the years one wants.

This program is valid from March 1, 1900 through February 28, 2100, but will generate erroneous output when the date supplied is February 29. The program will essentially consider every year as a leap year (having a February 29). The result is an output where every fourth year listed actually has a February 29 of the proper M -day. The user must know, then, at least one leap year in the sequence and count every fourth year from it in either direction. Also, the program does not test input for non-sensical dates and hence will produce correspondingly incorrect results.

Example 1:

When starting in 1949, is May 11 a Sunday?

Keystrokes:

Display:

[XEQ] [ALPHA] SIZE [ALPHA] 001

[XEQ] [ALPHA] M-DAY [ALPHA]

5[R/S]

11 [R/S]

0[R/S]

1949[R/S]

[R/S]

[R/S]

[R/S]

.

.

.

MONTH ?

DAY ?

M-DAY ?

START.YEAR ?

1952

1958

1969

1975

.

.

.

Program Listings

01♦LBL "M-D AY"	Prompting for month	49 ENTER↑	
02 "MONTH ?		50 ENTER↑	
"		51 GTO 08	
03 PROMPT		52♦LBL 00	
04 0		53 4	
05 STO 01		54 +	
06 CLX		55 RTN	
07 11	Day of week	56♦LBL 08	Prompt for
08 +	correction	57 "START.	starting year
09 14	factor	YEAR ?"	
10 X>Y?		58 PROMPT	
11 GTO 03		59 1	
12 RDN		60 -	
13♦LBL 01		61 ENTER↑	
14 2.6		62 R↑	
15 *	[2.6m]	63 -	
16 INT		64 28	
17 GTO 04		65 /	
18♦LBL 03		66 INT	
19 1		67 28	
20 STO 01		68 *	
21 ENTER↑		69 R↑	
22 +		70 RCL 01	
23 -		71 +	
24 +		72♦LBL 02	Addition Loop
25 GTO 01		73 +	
26♦LBL 04		74 X>Y?	
27 "DAY ?"	Prompt for day	75 XEQ 05	
28 PROMPT		76 6	
29 +		77 +	
30 "M-DAY ?	Prompt for M-day	78 X>Y?	
"		79 XEQ 05	
31 PROMPT		80 11	
32 -		81 +	
33 ENTER↑		82 X>Y?	
34 ENTER↑		83 XEQ 05	
35 7		84 6	
36 /		85 +	
37 INT		86 X>Y?	
38 7	$N = ([2.6m] + d - w)$	87 XEQ 05	
39 *	MOD]	88 5	
40 -		89 GTO 02	
41 2	N/Z	90♦LBL 05	
42 /		91 FIX 0	
43 ENTER↑		92 CF 29	
44 INT		93 CLA	
45 X≠Y?	If N is odd	94 ARCL X	
46 XEQ 00	add 4	95 FIX 4	
47 4		96 SF 29	
48 *	Year # MOD 28	97 PROMPT	
		98 RTN	

DATA REGISTERS				STATUS			
00		50		SIZE 002	TOT. REG. 25	USER MODE	
	USED			ENG	FIX	SCI	ON OFF X
				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					
40		90		ASSIGNMENTS			
				FUNCTION	KEY	FUNCTION	KEY
45		95					

IN WHAT YEAR IS A GIVEN DATE
AN M-DAY?
PROGRAM REGISTERS NEEDED: 24

ROW 1 (1 - 2)



ROW 2 (2 - 9)



ROW 3 (10 - 18)



ROW 4 (19 - 27)



ROW 5 (27 - 31)



ROW 6 (32 - 44)



ROW 7 (45 - 54)



ROW 8 (55 - 57)



ROW 9 (57 - 66)



ROW 10 (67 - 76)



ROW 11 (77 - 84)



ROW 12 (85 - 92)



ROW 13 (93 - 99)



NUMBER OF M-DAYS BETWEEN TWO DATES, AND THE N-TH M-DAY OF THE MONTH

In this program the user specifies a day-of-week and two dates; the program gives the number of times that day occurs between the given dates. User specifies a year, a month, a day-of-week, and a number N; the program computes the N-th occurrence of the specified day in the given month.

This program uses dates coded mm.ddyyyy. Program computes the number of M-days since a certain date in antiquity, using the formula:

$$N(m,d,y,w) = \left\lfloor \frac{D(m,d,y,w)}{7} \right\rfloor + 1/2 \left\lfloor 0.11 (D(m,d,y,w) \bmod 7) + 0.9 \right\rfloor$$

$$\text{where } D(m,d,y,w) = d - \left\lfloor 3/4 \left(\left\lfloor \frac{g(y,m)}{100} \right\rfloor - 7 \right) \right\rfloor + \left\lfloor 365.25 g(y,m) \right\rfloor + \left\lfloor 30.6 f(m) \right\rfloor - w$$

$$f(m) = \begin{cases} m+13 & m=1 \text{ or } 2 \\ m+1 & m > 2 \end{cases} \quad g(y,m) = \begin{cases} y-1 & m=1 \text{ or } 2 \\ y & m > 2 \end{cases} \quad \left\lfloor \right\rfloor = \text{INT}$$

W = M-Day (0 = Sunday, ..., 6 = Saturday)

The program then calculates the difference between the current N and the previous N.

Program first computes the first M-Day of the given month; given month = m year = y M-day = W (coded 0 = Sunday, ..., 6 = Saturday), the function computed is:

$$D(m,y,w) = h(W - \left\lfloor 2.6 f(m) \right\rfloor - \left\lfloor 5/4 g(y,m) \right\rfloor + \left\lfloor 3/4 \left(\left\lfloor \frac{g(y,m)}{100} \right\rfloor - 7 \right) \right\rfloor)$$

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

$$h(x) = \begin{cases} 7 & \text{if } x \bmod 7 = 0 \\ x \bmod 7 & \text{if } x \bmod 7 \neq 0 \end{cases}$$

For the N-TH M-Day the appropriate multiple of seven is added to the day-of-month.

This program is valid from the beginning of the Gregorian calendar through the year 9999.

Example 1:

How many Saturdays are there between July 4,1776 and July 4,1976?

Keystrokes:

Display:

[XEQ] [ALPHA] SIZE [ALPHA] 005

[XEQ] [ALPHA] NDAYS [ALPHA]

6 [R/S]

7.041776 [R/S]

7.041976 [R/S]

M-DAY ?

DATE 1 ?

DATE 2 ?

10436 TIMES

Example 2:

What is the second Wednesday of June, 1984?

Keystrokes:

Display:

[XEQ] [ALPHA] NTHDAY [ALPHA]

6 [R/S]

1984 [R/S]

3 [R/S]

2 [R/S]

MONTH ?

YEAR ?

M-DAY ?

NUM. ?

6-13-1984

User Instructions

				SIZE: 005
STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1.	Load program			
2.	Begin program execution		[XEQ] NDAYS	M-DAY ?
3.	Input day of week: 0 to 6	M-DAY	[R/S]	DATE 1 ?
4.	Input first date	DATE 1	[R/S]	DATE 2 ?
5.	Input second date	DATE 2	[R/S]	XXX TIMES
	For second option			
6.	Begin program execution		[XEQ] NTHDAY	MONTH ?
7.	Enter month	m	[R/S]	YEAR ?
8.	Enter year	y	[R/S]	M-DAY ?
9.	Enter day	d	[R/S]	NUM. ?
10.	Enter which occurrence	N	[R/S]	M-DD-YYYY

Program Listings

01♦LBL "NDA YS"	Prompt for day. and storage	46 -	
02 "M-DAY ? "		47 .75	
03 PROMPT		48 *	century days yy.yy
04 STO 04		49 INT	
05 "DATE 1 ?"	Prompt for first date.	50 X<>Y	
06 PROMPT	XEQ calculate sub.	51 36525	
07 XEQ 00	Prompt for sec- ond date.	52 *	[365.25y]
08 "DATE 2 ?"	XEQ calculate sub.	53 INT	
09 PROMPT		54 -	
10 XEQ 00		55 -	
11 FIX 0		56 INT	
12 CF 29		57 RCL 01	
13 CLA		58 30.6	[30.6 f(m)]
14 ARCL X		59 *	
15 "T TIMES "	Preparing dis- play and output	60 INT	
16 RVIEW		61 +	adjusted year recall day
17 FIX 4		62 RCL 04	
18 SF 29		63 -	
19 STOP		64 7	
20 GTO "NDA YS"		65 /	
21♦LBL 00		66 ENTER↑	
22 9		67 FRC	
23 ENTER↑		68 -	[2.6 f(m)]
24 1 E-6		69 LASTX	
25 -		70 .77	w-[2.6f(m)]
26 X<>Y		71 *	
27 3		72 .9	
28 X<>Y		73 +	
29 X<=Y?		74 INT	stack will fill with .07's
30 XEQ 09	If M=1 or 2 add 12 to M and Y← Y-1	75 2	
31 1		76 /	
32 +	Add 1 to M	77 +	
33 ENTER↑		78 RCL 02	w-month days- year days
34 INT		79 X<>Y	
35 STO 01	Store adjusted M	80 STO 02	
36 -		81 -	
37 1 E2		82 CHS	
38 *	.ddyyyy	83 RTN	
39 ENTER↑		84♦LBL 09	
40 FRC		85 +	
41 1 E2		86 +	
42 *	dd.yyyy	87 RTN	
43 ENTER↑		88♦LBL 02	
44 INT	YY.YY	89 RDN	
45 7	C=#of centuries	90 GTO 01	
		91♦LBL "NTH DAY"	
		92 "MONTH ? "	Prompt for month and store
		93 PROMPT	
		94 STO 03	

Program Listings

95 0		145 +	
96 STO 02	D(M,Y,W)/100	146 RCL 04	
97 CLX		147 1 E6	
98 1	recall year	148 /	
99 +		149 +	
100 4		150 RCL 03	
101 X<=Y?		151 +	
102 GTO 02	.ddyyyy	152 STO 03	
103 -1		153 FIX 6	
104 STO 02	mm.ddyyyy	154 "NUM. ?"	Prompt for
105 CLX		155 PROMPT	occurrence
106 8		156 1	
107 +		157 -	
108 +		158 .07	
109 LBL 01		159 *	
110 STO 01	Prompt for year	160 RCL 03	Calcualte date
111 "YEAR ?"		161 +	
112 PROMPT		162 FIX 0	
113 STO 04		163 CF 29	
114 ST+ 02		164 INT	
115 "M-DAY ?"	Prompt for day	165 CLA	
"		166 ARCL X	
116 PROMPT		167 LASTX	
117 RCL 01		168 FRC	
118 2.6		169 1 E2	
119 *		170 *	
120 INT		171 INT	
121 -		172 "f--"	
122 5		173 ARCL X	
123 RCL 02		174 LASTX	
124 .07		175 FRC	
125 RDN		176 1 E4	
126 *		177 *	
127 4		178 RND	
128 /		179 "f--"	
129 INT		180 ARCL X	
130 -		181 RVIEW	Display output
131 RCL 02		182 STOP	
132 1 E2		183 GTO "NTH	
133 /		DAY"	
134 INT		184 .END.	
135 7			
136 -			
137 .75			
138 *			
139 INT			
140 +			
141 7			
142 /			
143 FRC			
144 *			
	00		

DATA REGISTERS				STATUS			
00		50		SIZE 005	TOT. REG. 52	USER MODE	
	f(m)			ENG	FIX	SCI	ON OFF X
	code # Date			DEG	RAD	GRAD	
	M						
	Day of week			FLAGS			
05		55		#	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					

NO. OF M-DAYS BET. TWO DATES
AND THE NTH M-DAY OF THE MONTH
PROGRAM REGISTERS NEEDED: 48

ROW 1 (1 - 2)



ROW 2 (2 - 5)



ROW 3 (5 - 8)



ROW 4 (8 - 14)



ROW 5 (15 - 18)



ROW 6 (19 - 24)



ROW 7 (24 - 33)



ROW 8 (34 - 42)



ROW 9 (43 - 51)



ROW 10 (51 - 59)



ROW 11 (60 - 70)



ROW 12 (71 - 82)



ROW 13 (83 - 91)



ROW 14 (91 - 92)



ROW 15 (92 - 103)



ROW 16 (103 - 111)



ROW 17 (111 - 115)



ROW 18 (115 - 124)



NO. OF M-DAYS BET. TWO DATES
AND THE NTH M-DAY OF THE MONTH

ROW 19 (124 - 134)



ROW 20 (135 - 145)



ROW 21 (146 - 154)



ROW 22 (154 - 160)



ROW 23 (161 - 169)



ROW 24 (169 - 176)



ROW 25 (177 - 183)



ROW 26 (183 - 184)



HOLIDAYS

This program calculates the date of Mothers' Day, Fathers' Day, Election Day, Thanksgiving Day, Washington's Birthday, Labor Day, Columbus Day, and Memorial Day.

It uses the following formulas:

CODED:

MD	Mother's Day	= May 14 - D(0,y)
FD	Father's Day	= June 21 - D(3,y)
ED	Election Day	= Nov. 8 - D(1,y)
TG	Thanksgiving	= Nov. 28 - D(5,y)
WB	Wash.'s Birthday	= Feb. 21 - D(2,y-1)
LD	Labor Day	= Sep. 7 - D(3,y)
CD	Columbus Day	= Oct. 14 - D(5,y)
MM	Memorial Day	= May 31 - D(2,y)

Where $D(x,y) = (x + \left\lfloor \frac{5}{4}y \right\rfloor - \left\lfloor \frac{3}{4}(1 + \left\lfloor \frac{y}{100} \right\rfloor) \right\rfloor) \bmod 7$
 $\left\lfloor \right\rfloor = \text{INT}$

This program is valid from the beginning of the Gregorian Calendar through the year 9999.

Example 1:

When is Labor Day in 1980?

Keystrokes:

Display:

[XEQ] [ALPHA] HOLDY [ALPHA]
 LD [R/S]
 1980 [R/S]

HOLIDAY ?
 YEAR ?
 9-1-1980

Example 2:

When was Washington's Birthday in the year 1900?

Keystrokes:

[XEQ] [ALPHA] HOLDY [ALPHA]

WB [R/S]

1900 [R/S]

Display:

HOLIDAY ?

YEAR ?

2-19-1900

Program Listings

01♦LBL "HOL DY"	Prompt for holi- day in alpha mode	49 INT	
02 "HOLIDAY ?"		50 1	
03 AON		51 +	Century year ad- justment
04 PROMPT		52 .75	
05 AOFF		53 *	
06 ASTO Y		54 INT	
07 "YEAR?"	Get year	55 -	
08 PROMPT		56 7	
09 GTO IND	Go to appropri- ate subroutine	57 /	
Z		58 FRC	
10♦LBL "MD"		59 .07	Days to be sub- tracted
11 0		60 *	
12 XEQ 09		61 CHS	
13 5.14	Mother's Day	62 X<>Y	
14 +		63 1 E6	Encore year
15 GTO 05		64 /	
16♦LBL "FD"	Father's Day	65 +	
17 3		66 RTN	
18 XEQ 09		67♦LBL "WB"	Washington's Birthday
19 6.21		68 1	
20 +		69 -	
21 GTO 05		70 2	
22♦LBL "ED"	Election Day	71 XEQ 09	
23 1		72 2.210001	
24 XEQ 09		73 +	
25 11.08		74 GTO 05	Labor Day
26 +		75♦LBL "LD"	
27 GTO 05		76 3	
28♦LBL "TG"	Thanksgiving Day	77 XEQ 09	
29 5		78 9.07	
30 XEQ 09		79 +	
31 11.28		80 GTO 05	
32 +		81♦LBL "CD"	Columbus Day
33 GTO 05		82 5	
34♦LBL 09	Main computation routine	83 XEQ 09	
35 X<>Y		84 10.14	
36 ENTER↑		85 +	
37 ENTER↑		86 GTO 05	
38 RDN		87♦LBL "MM"	Memorial Day
39 RDN		88 2	
40 5		89 XEQ 09	
41 *		90 5.31	
42 4		91 +	
43 /		92♦LBL 05	Output routine
44 INT		93 INT	
45 +		94 CLA	
46 X<>Y		95 FIX 0	
47 1 E2	C=Number of centuries	96 CF 29	Prepare date for output
48 /		97 ARCL X	
		98 LASTX	
		99 FRC	

Program Listings

100 1 E2	Output date	51	
101 *			
102 INT			
103 "I--"			
104 ARCL X			
105 LASTX			
106 FRC			
107 1 E4			
108 *			
109 RND		60	
110 "I--"			
111 ARCL X			
112 AVIEW			
113 SF 29			
114 FIX 4			
115 END			
20		70	
30		80	
40		90	
50		00	

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS				STATUS				
00		50		SIZE	000	TOT. REG.	37	USER MODE
				ENG		FIX	0,4	SCI
				DEG		RAD		GRAD
								ON
								OFF
05		55		FLAGS				
				#	INIT S/C	SET INDICATES	CLEAR INDICATES	
				29		Digit separators	Separators omitted	
10		60						
15		65						
20		70						
25		75						
30		80						
35		85						
				ASSIGNMENTS				
				FUNCTION		KEY	FUNCTION	
							KEY	
40		90						
45		95						

HOLIDAYS

PROGRAM REGISTERS NEEDED: 37

ROW 1 (1: 2)



ROW 2 (2: 7)



ROW 3 (7: 10)



ROW 4 (11: 16)



ROW 5 (16: 20)



ROW 6 (21: 25)



ROW 7 (25: 28)



ROW 8 (29: 34)



ROW 9 (35: 47)



ROW 10 (47: 56)



ROW 11 (57: 65)



ROW 12 (66: 71)



ROW 13 (72: 75)



ROW 14 (75: 79)



ROW 15 (80: 84)



ROW 16 (84: 87)



ROW 17 (88: 95)



ROW 18 (95: 103)



HOLIDAYS

HEWLETT PACKARD
SOLUTIONS BOOK
CALENDARS

ROW 19 (103 : 110)



ROW 20 (110 : 115)



RELIGIOUS HOLIDAYS

This program calculates the dates of various holidays as listed below:

<u>CODE:</u>	<u>HOLIDAY:</u>
AW	ASH WEDNESDAY
FS	FIRST SUNDAY IN LENT
PA	PASSION SUNDAY
PM	PALM SUNDAY
GF	GOOD FRIDAY
RS	ROGATION SUNDAY
AD	ASCENSION DAY
WS	WHITSUNDAY
TS	TRINITY SUNDAY
EA	EASTER

Easter falls on the first Sunday following the arbitrary Paschal Full Moon, which does not necessarily coincide with a real or astronomical full moon. The Golden Number is calculated by adding 1 to the remainder obtained by dividing the year by 19. The Paschal Full Moon is determined by applying the Golden Number to the following table:

<u>Golden</u> <u>Number</u>	<u>Date</u>	<u>Golden</u> <u>Number</u>	<u>Date</u>	<u>Golden</u> <u>Number</u>	<u>Date</u>	<u>Golden</u> <u>Number</u>	<u>Date</u>	<u>Golden</u> <u>Number</u>	<u>Date</u>
1	Apr 14	5	Mar 31	9	Apr 16	13	Apr 2	17	Apr 17
2	Apr 3	6	Apr 18	10	Apr 5	14	Mar 22	18	Apr 7
3	Mar 23	7	Apr 8	11	Mar 25	15	Apr 10	19	Mar 27
4	Apr 11	8	Mar 28	12	Apr 13	16	Mar 30		

If the Paschal Full Moon falls on a Sunday, Easter is on the following Sunday. The earliest Easter can fall is March 23rd and the latest is April 25th. Ash Wednesday and the other religious holidays are determined by specific number of days from Easter. This program is good for the 200-year period of March 1, 1900 through February 28, 2100.

Example 1:

Determine when Easter occurs during the year of 1980?

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 012

[XEQ] [ALPHA] RL [ALPHA]

1980 [R/S]

EA [R/S]

Display:

YR?

HOL?

5. (Golden Number)

4.061980 (Apr. 6, 1980)

Example 2:

Determine when Whitsunday occurred in 1968?

Keystrokes:

[XEQ] [ALPHA] RL [ALPHA]

1968 [R/S]

WS [R/S]

Display:

YR?

HOL?

12. (Golden Number)

6.021968 (June 2, 1968)

[illegible]

Program Listings

01♦LBL "RL"	Initialization and input of variables	48 .4	"Packed Register to find Paschal Full Moon
02 365.25		49 X<>Y	
03 STO 10		50 X=Y?	
04 30.6001		51 GTO 02	
05 STO 11		52 .6	
06 122.1		53 X=Y?	
07 STO 05		54 GTO 03	
08 241302.2		55 1	
1		56 GTO 05	
09 STO 00		57♦LBL 00	
10 10281807		58 1 E8	Determine following Sunday
.26		59 GTO 05	
11 STO 01		60♦LBL 01	
12 15042312		61 1 E6	
.01		62 GTO 05	
13 STO 02		63♦LBL 02	
14 20092717		64 1 E4	
.06		65 GTO 05	
15 STO 03		66♦LBL 03	
16 "YR?"		67 1 E2	
17 PROMPT		68♦LBL 05	Go to Holiday Ash Wednesday First Sunday in Lent Passion Sunday
18 STO 09		69 RCL IND	
19 "HOL?"		06	
20 AON		70 X<>Y	
21 PROMPT		71 /	
22 AOFF		72 FRC	
23 RCL 10		73 1 E2	
24 *		74 *	
25 INT	Determine Golden number and dis- play	75 INT	
26 143		76 ST+ 04	
27 +		77 RCL 04	Go to Holiday Ash Wednesday First Sunday in Lent Passion Sunday
28 STO 04		78 5	
29 RCL 09		79 +	
30 19		80 7	
31 MOD		81 MOD	
32 1		82 FIX 0	
33 +		83 RND	
34 FIX 0		84 7	
35 PSE		85 -	
36 5		86 ST- 04	
37 /		87 ASTO X	Go to Holiday Ash Wednesday First Sunday in Lent Passion Sunday
38 FIX 1	Select appro- priate packed register	88 GTO IND	
39 RND		X	
40 STO 06		89♦LBL "AW"	
41 FRC		90 -46	
42 X=0?		91 GTO 04	
43 GTO 00		92♦LBL "FS"	
44 .2		93 -42	
45 X<>Y		94 GTO 04	
46 X=Y?		95♦LBL "PA"	
47 GTO 01	Deciphering	96 -14	

Program Listings

97 GTO 04		148 -	
98♦LBL "PM"	Palm Sunday	149 -	
99 -7		150 RCL 07	
100 GTO 04		151 14	
101♦LBL "GF"	Good Friday	152 /	
102 -2		153 INT	
103 GTO 04		154 ST+ 09	
104♦LBL "RS"	Rogation Sunday	155 12	
105 35		156 *	
106 GTO 04		157 -	
107♦LBL "AD"	Ascension Day	158 RCL 09	
108 39		159 1 E6	
109 GTO 04		160 /	
110♦LBL "WS"	Whitsunday	161 FIX 6	
111 49		162 +	
112 GTO 04		163 END	
113♦LBL "TS"	Trinity Sunday		
114 56			
115♦LBL 04			
116 ST+ 04			
117♦LBL "EA"	Easter		
118♦LBL 04			
119 RCL 04			
120 RCL 05			
121 -			
122 RCL 10	Decipher day		
123 /	number into		
124 INT	Gregorian date		
125 STO 09			
126 RCL 10			
127 *			
128 INT			
129 RCL 04			
130 -			
131 CHS			
132 STO 06			
133 RCL 11			
134 /			
135 INT			
136 STO 07			
137 RCL 06			
138 X<>Y			
139 RCL 11			
140 *			
141 INT			
142 -			
143 STO 08			
144 RCL 07	Get and format		
145 1	output		
146 RCL 08			
147 %			

[illegible]

RELIGIOUS HOLIDAYS

PROGRAM REGISTERS NEEDED: 49

ROW 1 (1 : 3)



ROW 2 (4 : 6)



ROW 3 (7 : 10)



ROW 4 (10 : 12)



ROW 5 (12 : 14)



ROW 6 (14 : 17)



ROW 7 (18 : 26)



ROW 8 (26 : 35)



ROW 9 (36 : 45)



ROW 10 (46 : 54)



ROW 11 (54 : 61)



ROW 12 (61 : 67)



ROW 13 (68 : 76)



ROW 14 (77 : 87)



ROW 15 (87 : 91)



ROW 16 (91 : 95)



ROW 17 (95 : 98)



ROW 18 (98 : 101)



RELIGIOUS HOLIDAYS

ROW 19 (102 : 106)



ROW 20 (106 : 110)



ROW 21 (110 : 113)



ROW 22 (113 : 118)



ROW 23 (119 : 131)



ROW 24 (132 : 144)



ROW 25 (145 : 155)



ROW 26 (155 : 163)



ROW 27 (163 : 163)



CHINESE YEARS TO/FROM GREGORIAN YEARS

This program converts the cyclical two-character year designation of the Chinese Calendar to or from Gregorian years.

The Chinese year designation consists of two characters, the "most significant bit" called (干) and the "least significant bit" called (支). There are 10 (干)'s and 12 (支)'s which are represented here by consecutive numbers for computation purposes. The rule states that any odd number of Group One can associate with only the odd numbers of Group Two, and likewise even numbers can associate with only even numbers. This way, 60 possible combinations form one basic cycle of 60 years. Each cycle begins with the same year (甲子), or (1, 1) in the numeralized notations, or (R:1 C:1) in our machine notations, which falls in the years 1924 and $1924 + 60N$; where N is any integer. This program arbitrarily selects ($N = -85$) as the internal datum but displays 1924 ($N = 0$) externally as the first year and 1983 as the 60th year of the machine cycle. Actually, the machine cycle may be set to start with a year corresponding to any N. The fact that the sum of the two numbers for any year is always an even number provides us with a parity check. For example, the year 1975 falls in row 2 and column 4 ($2+4=6$), and the year is (R:2 C:4) or (乙卯).

		子 1	丑 2	寅 3	卯 4	辰 5	巳 6	午 7	未 8	申 9	酉 10	戌 11	亥 12
甲	1	1924		1974		1964		1954		1944		1934	
乙	2		1925		1975		1965		1955		1945		1935
丙	3	1936		1926		1976		1966		1956		1946	
丁	4		1937		1927		1977		1967		1957		1947
戊	5	1948		1938		1928		1978		1968		1958	
己	6		1949		1939		1929		1979		1969		1959
庚	7	1960		1950		1940		1930		1980		1970	
辛	8		1961		1951		1941		1931		1981		1971
壬	9	1972		1962		1952		1942		1932		1982	
癸	10		1973		1963		1953		1943		1933		1983

Example 1:

Convert the year 1956 to its Chinese equivalent.

Keystrokes:	Display:	
[USER]		(Set USER mode)
[XEQ] [ALPHA] SIZE [ALPHA] 009		
[XEQ] [ALPHA] G-C [ALPHA]	YEAR ?	
1956 [R/S]	R:3 C:9	(Row 3, Column 9)

Example 2:

Convert the Chinese year Row-2 and Column 4 (乙卯) to Gregorian year.

Keystrokes:	Display:
[XEQ] [ALPHA] C-G [ALPHA]	ROW?
2 [R/S]	COLUMN?
4 [R/S]	1975

Program Listings

01♦LBL "G-C		49 -COLUMN	Prompt for column
"		?"	
02 FIX 0	Initialization	50 PROMPT	
03 CF 29		51 X>Y?	
04 CLRG		52 XEQ 00	
05 -3176		53 STO 07	
06 STO 01		54 RCL 06	
07 10		55 RCL 07	
08 STO 02		56 -	
09 12		57 2	
10 STO 03		58 /	
11 FS?C 05		59 STO 08	
12 RTN		60 FRC	
13 "YEAR ?"	Prompt for year	61 0	
14 PROMPT		62 X*Y?	Parity check
15 RCL 01		63 GTO 01	
16 -		64 RCL 08	
17 STO 05	ΔY	65 RCL 03	Compute years in excess of datum year
18 RCL 05		66 *	
19 RCL 02		67 RCL 07	
20 /		68 +	
21 INT		69 1	
22 RCL 02		70 -	
23 *		71 1924	
24 -		72 +	
25 1	The most significant	73 VIEW X	
26 +		74 SF 29	
27 RCL 05		75 FIX 4	
28 RCL 05		76 RTN	
29 RCL 03		77♦LBL C	
30 /		78 60	60, +
31 INT		79 +	
32 RCL 03		80 RTN	
33 *		81♦LBL D	
34 -	The least significant	82 60	60, -
35 1		83 -	
36 +		84 RTN	
37 " R:"		85♦LBL 00	
38 ARCL Y	Display	86 RCL 03	
39 "F C:"		87 -	
40 ARCL X		88 RTN	
41 AVIEW		89♦LBL 01	
42 STOP		90 "ILLEGAL DATE"	Illegal date message
43♦LBL "C-G		91 AVIEW	
"	Initialize	92 STOP	
44 SF 05		93 .END.	
45 XEQ "G-C			
"			
46 "ROW ?"			
47 PROMPT	Prompt for row		
48 STO 06		00	

[illegible]

CHINESE YEARS TO/FROM
GREGORIAN YEARS
PROGRAM REGISTERS NEEDED: 26

ROW 1 (1 - 5)



ROW 2 (5 - 11)



ROW 3 (12 - 18)



ROW 4 (19 - 31)



ROW 5 (32 - 39)



ROW 6 (39 - 43)



ROW 7 (43 - 46)



ROW 8 (46 - 49)



ROW 9 (49 - 59)



ROW 10 (60 - 71)



ROW 11 (71 - 77)



ROW 12 (78 - 87)



ROW 13 (88 - 90)



ROW 14 (90 - 93)



NEW MOON AND FULL MOON DAY OF MONTH

This program follows an empirical algorithm published in one issue of the amateur astronomers monthly, SKY & TELESCOPE. The author of the article, O.L. Harvey claims, "By testing hundreds of dates across the interval from 1001 B.C. to A.D. 2003, I found that about half the answers obtained with this device agree exactly with the day of new moon, and about half are off by one day. A very small number (two percent in my sample) are two days in error." Mr. Harvey developed a three part table based on repeating patterns in an enormous data source and on averages of date differences. An entry point was calculated for the first part, the resultant was the entry point for the second part. A calculated difference on the second part produced a number. The month provided the key for the third part of the table. These numbers are combined with other calculations to determine the day of month. Mr. Vanderburgh devised an HP-65 program to calculate the entry points and when used with the table and additional inputs would complete the calculations for the day of month.

This program uniquely incorporates that table into the registers and selects the appropriate numbers to perform the complete calculations for day of month.

This program also requires an additional memory module.

Example 1:

Determine the date of the new moon in November, 1979.

Keystrokes:	Display:
[XEQ] [ALPHA] SIZE [ALPHA] 026	
[XEQ] [ALPHA] MOONS [ALPHA]	N OR F MOON?
N [R/S]	MM.YYYY ?
11.1979 [R/S]	11-18-1979

Example 2:

Find the date of the full moon in January of 1956.

Keystrokes:	Display:
[XEQ] [ALPHA] MOON [ALPHA]	N OR F MOON?
F [R/S]	MM.YYYY ?
1.1956 [R/S]	1-27-1956

[illegible]

Program Listings

01♦LBL "MOO NS" 02 .0925102 612 03 STO 00 04 .2813291 501 05 STO 01 06 .1703190 420 07 STO 02 08 .0622072 3 09 STO 03 10 .1113000 2 11 STO 04 12 .0204040 6 13 STO 05 14 .0708091 0 15 STO 06 16 .301020 17 STO 10 18 .112101 19 STO 11 20 .220212 21 STO 12 22 .031323 23 STO 13 24 .142404 25 STO 14 26 .250515 27 STO 15 28 .061626 29 STO 16 30 .172707 31 STO 17 32 .280818 33 STO 18 34 .091929 35 STO 19 36♦LBL "MOO N" 37 "F" 38 ASTO Y 39 "N OR F MOON?" 40 AON 41 PROMPT	Initialization	42 AOFF 43 ASTO X 44 X=Y? 45 SF 05 46 "MM.YYYY ?" 47 PROMPT 48 CLA 49 FIX 4 50 INT 51 STO 20 52 LASTX 53 FRC. 54 STO 22 55 1 E4 56 * 57 STO 21 58 2 59 RCL 20 60 X<=Y? 61 DSE 21 62♦LBL 00 63 RCL 21 64♦LBL 01 65 0 66 STO 07 67 1582 68 RCL 21 69 X<=Y? 70 GT0 03 71 13 72 STO 07 73 1900 74 STO 23 75♦LBL 02 76 RCL 21 77 RCL 23 78 X<=Y? 79 GT0 03 80 1 E2 81 - 82 STO 23 83 1 84 ST- 07 85 GT0 02 86♦LBL 03 87 RCL 21 88 4712 89 + 90 76 91 /	Prompt for year	Check for Jan. or Feb.	Check for year before 1582	Check for year after 1582	Calculate Julian year
--	----------------	---	-----------------	---------------------------	-------------------------------	------------------------------	--------------------------

Program Listings

92 INT		143 2	
93 STO 23		144 ST+ 09	
94 LASTX		145 XEQ 09	
95 FRC		146♦LBL 08	Count down in
96 76		147 RCL 25	table L by
97 *		148 RCL 08	difference
98 RND		149 -	
99 4		150 10	
100 /		151 X>Y?	
101 INT	Entry point to	152 GTO 06	
102 STO 24	table R	153 RDN	
103 LASTX		154 STO 25	
104 FRC		155♦LBL 07	Number from
105 4		156 XEQ 09	table L
106 *		157 ST+ 07	
107 STO 08		158 RCL 20	
108 RCL 23	Difference in	159 1	
109 4	table L	160 -	
110 /		161 4	
111 INT		162 /	
112 ST- 07		163 4	
113 RCL 24		164 +	
114 5		165 STO 25	
115 /		166 FRC	
116 STO 25		167 8	
117 FRC		168 *	
118 10		169 STO 09	
119 *		170 XEQ 09	
120 STO 09	Number from	171 ST- 07	Number from
121 XEQ 09	table R	172 FS?C 05	table M
122 STO 24		173 XEQ 05	
123 10		174 RCL 07	
124 /		175 30	
125 FRC		176 X<=Y?	Check date for
126 10		177 -	range 0 and 30
127 *		178 0	
128 LASTX		179 RCL 07	
129 +		180 X>Y?	
130 STO 25		181 GTO 04	
131 0		182 30	
132 STO 09		183 RCL 07	
133 XEQ 09		184 +	
134 RCL 24		185 GTO 04	
135 X=Y?	Search for	186♦LBL 09	
136 GTO 08	number from	187 RCL 09	
137 2	table R in table	188 10↑X	Obtain appro-
138 ST+ 09	L	189 RCL IND	priate number
139 XEQ 09		25	from selected
140 RCL 24		190 *	register
141 X=Y?		191 FRC	
142 GTO 08		192 1 E2	

Program Listings

193 *		51	
194 INT			
195 RTN			
196♦LBL 06			
197 +	Check for re- turn to table L		
198 STO 25			
199 5			
200 RCL 09			
201 2			
202 +		60	
203 X>Y?			
204 0			
205 STO 09			
206 GTO 07			
207♦LBL 04			
208 1 E2			
209 /			
210 RCL 22	Put output into proper display format: MM-DD-YYYY		
211 LASTX			
212 /		70	
213 +			
214 RCL 20			
215 +			
216 FIX 0			
217 CF 29			
218 INT			
219 ARCL X			
220 "┐—"			
221 LASTX			
222 FRC			
223 1 E2		80	
224 *			
225 INT			
226 ARCL X			
227 "┐—"			
228 LASTX			
229 FRC			
230 1 E4			
231 *			
232 ARCL X			
233 AVIEW		90	
234 RTN			
235♦LBL 05			
236 15			
237 ST+ 07			
238 RTN			
239 .END.			
50		00	

[illegible]

NEW MOON AND FULL MOON
DAY OF MONTH
PROGRAM REGISTERS NEEDED: 73

ROW 1 (1 - 2)



ROW 2 (2 - 4)



ROW 3 (4 - 6)



ROW 4 (6 - 8)



ROW 5 (8 - 11)



ROW 6 (12 - 14)



ROW 7 (14 - 16)



ROW 8 (16 - 20)



ROW 9 (20 - 23)



ROW 10 (24 - 26)



ROW 11 (26 - 30)



ROW 12 (30 - 32)



ROW 13 (32 - 35)



ROW 14 (36 - 39)



ROW 15 (39 - 40)



ROW 16 (41 - 46)



ROW 17 (46 - 53)



ROW 18 (54 - 61)



NEW MOON AND FULL MOON
DAY OF MONTH

ROW 19 (61 - 68)



ROW 20 (69 - 75)



ROW 21 (76 - 82)



ROW 22 (83 - 89)



ROW 23 (90 - 99)



ROW 24 (100 - 110)



ROW 25 (111 - 119)



ROW 26 (120 - 127)



ROW 27 (128 - 136)



ROW 28 (136 - 143)



ROW 29 (144 - 151)



ROW 30 (152 - 158)



ROW 31 (159 - 170)



ROW 32 (170 - 176)



ROW 33 (177 - 186)



ROW 34 (187 - 196)



ROW 35 (197 - 207)



ROW 36 (208 - 216)



NEW MOON AND FULL MOON
DAY OF MONTH

ROW 37 (216 - 223)



ROW 38 (223 - 230)



ROW 39 (231 - 239)



ROW 40 (239 - 239)



CALENDAR PRINTOUT

This program will printout a calendar for any valid Gregorian year between March, 1900 and February, 2100. This program requires a printer and two memory modules.

NOTE: If any discrepancy is noticed, see the reference for further details.

REFERENCE: O'Neil, W. M., Time and the Calendars, Sydney Univ. Press, Australia, 1975.

Example 1:

Print a calendar for the months June, July and August of the year 1976.

Keystrokes:

[XEQ] [ALPHA] SIZE [ALPHA] 043

[XEQ] [ALPHA] INIT [ALPHA]

1976 [R/S]

6 [R/S]

3 [R/S]

Display:

YEAR ?

START. MONTH?

NO. OF MON. ?

```

                JUNE
                1976
S M T W T F S
   1  2  3  4  5
  6  7  8  9 10 11 12
 13 14 15 16 17 18 19
 20 21 22 23 24 25 26
 27 28 29 30
  
```

(Printout)

```

                JULY
                1976
S M T W T F S
           1  2  3
  4  5  6  7  8  9 10
 11 12 13 14 15 16 17
 18 19 20 21 22 23 24
 25 26 27 28 29 30 31
  
```

```

                AUGUST
                1976
S M T W T F S
   1  2  3  4  5  6  7
   8  9 10 11 12 13 14
 15 16 17 18 19 20 21
 22 23 24 25 26 27 28
 29 30 31
  
```

Example 2:

Print the calendar for the first 4 months of the year 1980.

Keystrokes:

[XEQ] [ALPHA] CAL [ALPHA]

1980 [R/S]

1 [R/S]

4 [R/S]

Display:

YEAR?

START. MONTH?

NO. OF MON.?

(Print Out)

JANUARY
1980
S M T W T F S
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

FEBRUARY
1980
S M T W T F S
1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29

MARCH
1980
S M T W T F S
1
2 3 4 5 6 7 8
9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31

APRIL
1980
S M T W T F S
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30

[illegible]

Program Listings

<pre> 01♦LBL "INT T" 02 CF 00 03 SF 21 04 1.031 05 STO 00 06♦LBL 00 07 9 08 RCL 00 09 INT 10 X<=Y? 11 " " 12 FIX 0 13 CF 29 14 ARCL X 15 ASTO IND X 16 CLA 17 ISG 00 18 GTO 00 19 0 20 STO 00 21 722923 22 STO 40 23 722892 24 STO 41 25 GTO "CAL .. </pre>	Initialize flags & registers	<pre> 49 32 50 BLDSPEC 51 8 52 RCL 33 53 - 54 1 E3 55 / 56 1 57 + 58♦LBL 13 59 ARCL Y 60 ARCL IND X 61 ISG X 62 GTO 13 63 ARCL Y 64 ACA 65 ADV 66 9 67 RCL 33 68 - 69 ENTER↑ 70 ENTER↑ 71 ENTER↑ 72 7 73 MOD 74 X=0? 75 SF 00 76 FS? 00 77 + 78 FS?C 00 79 7 80 RCL 34 81 - 82 CHS 83 1 84 + 85 7 86 / 87 INT 88 7 89 * 90 .07 91 + 92 1 E3 93 / 94 + 95 STO 00 96♦LBL 23 97 RCL 00 </pre>	Loop to complete first line of the calendar Print line of calendar End of the line? If so, move to next line of calendar Otherwise, cont. with same line. Build & print middle lines
<pre> 26♦LBL 16 27 RCL 33 28 1 29 - 30 1 E3 31 / 32 1 33 + 34 0 35 ENTER↑ 36 32 37 BLDSPEC 38 CLA 39 ARCL X 40 "T " 41 ASTO X 42 CLA 43♦LBL 14 44 ARCL X 45 ISG Y 46 GTO 14 47 0 48 ENTER↑ </pre>	Routine to build and print the lines in the calendar Get value for DOW of 1st day Loop to blank - fill portion of first line		

Program Listings

98 INT	of the	146 RCL 35	
99 ENTER↑	calender	147 -	
100 ENTER↑		148 1 E3	
101 6		149 /	
102 +		150 1	
103 1 E3		151 +	
104 /		152♦LBL 18	
105 +		153 ARCL Y	Loop to fill in
106 0		154 "F "	blanks at end of
107 ENTER↑		155 ISG X	last line
108 32		156 GTO 18	
109 BLDSPEC		157 0	
110 CLA		158 ENTER↑	
111♦LBL 22		159 32	
112 ARCL X	Loop to build	160 BLDSPEC	
113 ARCL IND	the middle lines	161 ARCL X	
Y		162 ACA	
114 ISG Y		163 ADV	
115 GTO 22		164 RTN	
116 ARCL X	Print the line	165♦LBL 29	Subroutine to
117 ACA		166 RCL 37	compute the
118 ADV		167 RCL 36	day of week
119 ISG 00		168 1	of the first
120 GTO 23		169 +	day of the
121 RCL 34		170 ENTER↑	month and the
122 RCL 00		171 1/X	Julian day
123 INT		172 .7	
124 X>Y?		173 +	
125 RTN		174 CHS	
126 ENTER↑		175 INT	
127 ENTER↑		176 ST+ 37	
128 RCL 35		177 12	
129 +		178 *	
130 1		179 -	
131 -		180 30.6001	
132 1 E3		181 *	
133 /		182 INT	
134 +		183 RCL 37	
135 0		184 365.25	
136 ENTER↑		185 *	
137 32		186 INT	
138 BLDSPEC		187 +	
139 CLA		188 STO 40	
140♦LBL 20		189 7	
141 ARCL X	Loop to build	190 MOD	
142 ARCL IND	last line	191 X=0?	
Y		192 7	
143 ISG Y		193 RTN	
144 GTO 20		194♦LBL "CAL	
145 7		"	

Program Listings

195 "YEAR ?"	Prompt for year	242 "M"	
196 PROMPT	and initial	243 ACA	
197 STO 38	month	244 SKPCOL	
198 "START.		245 "T"	
MONTH?"		246 ACA	
199 PROMPT		247 SKPCOL	
200 ENTER↑		248 "W"	
201 ENTER↑		249 ACA	
202 "NO. OF	Ask for how many	250 SKPCOL	
MON. ?"	months to print	251 "T"	
203 PROMPT		252 ACA	
204 +		253 SKPCOL	
205 1		254 "F"	
206 -		255 ACA	
207 .001		256 SKPCOL	
208 *		257 "S"	
209 +		258 ACA	
210 STO 42		259 2	
211♦LBL 15	Set up counter	260 SKPCOL	
212 RCL 42		261 ADV	
213 INT		262 CF 12	
214 STO 36		263 XEQ 16	Print the
215 RCL 38		264 ADV	calendar
216 STO 37		265 ADV	
217 XEQ 29		266 RCL 42	
218 STO 33	DOW of 1st day	267 INT	If month is
219 RCL 40	of month	268 12	December then
220 STO 41	Save Julian	269 X=Y?	fix the pointers
221 1	day	270 XEQ 30	
222 ST+ 36		271 ISG 42	
223 RCL 38		272 GTO 15	
224 STO 37		273 RTN	
225 XEQ 29		274♦LBL 01	
226 1		275 " JANU	
227 -		ARY"	
228 STO 35	DOW of last day	276 AVIEW	
229 RCL 40	of month	277 RTN	
230 RCL 41		278♦LBL 02	
231 -		279 " FEBR	
232 STO 34	# days in the	UARY"	
233 SF 12	month	280 AVIEW	
234 XEQ IND	Print month name	281 RTN	
42		282♦LBL 03	
235 " "	Print the year	283 " MAR	
236 ARCL 38		CH"	
237 AVIEW		284 AVIEW	
238 "S"		285 RTN	
239 ACA	Print DOW labels	286♦LBL 04	
240 4		287 " APR	
241 SKPCOL		IL"	
		288 AVIEW	

Program Listings

289 RTN			
290♦LBL 05			
291 " MA			
Y"			
292 AVIEW			
293 RTN			
294♦LBL 06			
295 " JU			
NE"			
296 AVIEW			
297 RTN			
298♦LBL 07			
299 " JU			
LY"			
300 AVIEW			
301 RTN			
302♦LBL 08			
303 " AUG			
UST"			
304 AVIEW			
305 RTN			
306♦LBL 09			
307 " SEPTE			
MBER"			
308 AVIEW			
309 RTN			
310♦LBL 11			
311 " NOVEM			
BER"			
312 AVIEW			
313 RTN			
314♦LBL 10			
315 " OCTO			
BER"			
316 AVIEW			
317 RTN			
318♦LBL 12			
319 " DECEM			
BER"			
320 AVIEW			
321 RTN			
322♦LBL 30			
323 1	Increment the		
324 ST+ 38	year		
325 12			
326 ST- 42	Set the month		
327 .001	# to 1		
328 *			
329 ST- 42	Fix pointer in		
330 RTN	R ₄₂		
331 .END.			

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS			STATUS			
00	USED FOR LOOP COUNTER	50	SIZE	043	TOT. REG.	142
	" 1"		ENG		FIX	SCI
	" 2"		DEG		RAD	GRAD
	" 3"		USER MODE			
	" 4"		ON OFF X			
05	" 5"	55	FLAGS			
	" 6"		#	INIT S/C	SET INDICATES	CLEAR INDICATES
	" 7"		00	C	Go to next cal row	Stay on same row
	" 8"		12		Print double wide	Print single width
10	"10"	60	21	S	Printer enable	Printer disable
	"11"		29	C	Radix point	No radix point
	"12"					
	"13"					
	"14"					
15	"15"	65				
	"16"					
	"17"					
	"18"					
	"19"					
20	"20"	70				
	"21"					
	"22"					
	"23"					
	"24"					
25	"25"	75				
	"26"					
	"27"					
	"28"					
	"29"					
30	"30"	80				
	"31"					
	USED					
	DOW OF 1st OF MONTH					
	# DAYS IN MONTH					
35	DOW OF LAST DAY OF MONTH	85				
	FIRST MONTH					
	YEAR		ASSIGNMENTS			
	YEAR					
	USED		FUNCTION	KEY	FUNCTION	KEY
40	JULIAN DAY #1	90				
	JULIAN DAY #2					
	MONTH NUMBER					
45		95				

CALENDAR PRINTOUT

PROGRAM REGISTERS NEEDED: 99

ROW 1 (1 : 4)



ROW 2 (4 : 12)



ROW 3 (12 : 19)



ROW 4 (20 : 23)



ROW 5 (23 : 27)



ROW 6 (28 : 37)



ROW 7 (37 : 44)



ROW 8 (44 : 52)



ROW 9 (52 : 60)



ROW 10 (61 : 68)



ROW 11 (69 : 78)



ROW 12 (79 : 90)



ROW 13 (90 : 98)



ROW 14 (99 : 108)



ROW 15 (109 : 115)



ROW 16 (115 : 121)



ROW 17 (122 : 132)



ROW 18 (132 : 141)



CALENDAR PRINTOUT

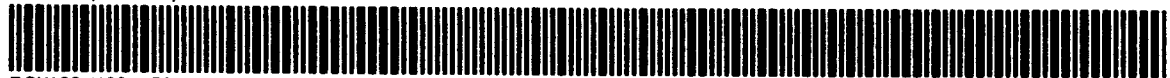
ROW 19 (141 : 148)



ROW 20 (148 : 154)



ROW 21 (155 : 161)



ROW 22 (162 : 170)



ROW 23 (171 : 180)



ROW 24 (180 : 184)



ROW 25 (184 : 193)



ROW 26 (194 : 195)



ROW 27 (195 : 198)



ROW 28 (198 : 202)



ROW 29 (202 : 206)



ROW 30 (207 : 213)



ROW 31 (214 : 219)



ROW 32 (220 : 226)



ROW 33 (227 : 234)



ROW 34 (234 : 239)



ROW 35 (239 : 246)



ROW 36 (246 : 252)



CALENDAR PRINTOUT

ROW 37 (253 : 259)



ROW 38 (260 : 267)



ROW 39 (268 : 274)



ROW 40 (275 : 277)



ROW 41 (278 : 279)



ROW 42 (280 : 283)



ROW 43 (284 : 287)



ROW 44 (288 : 292)



ROW 45 (293 : 296)



ROW 46 (297 : 300)



ROW 47 (301 : 303)



ROW 48 (304 : 307)



ROW 49 (307 : 311)



ROW 50 (311 : 315)



ROW 51 (315 : 319)



ROW 52 (319 : 325)



ROW 53 (325 : 331)



ROW 54 (331 : 331)



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

Original content used with permission.

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

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