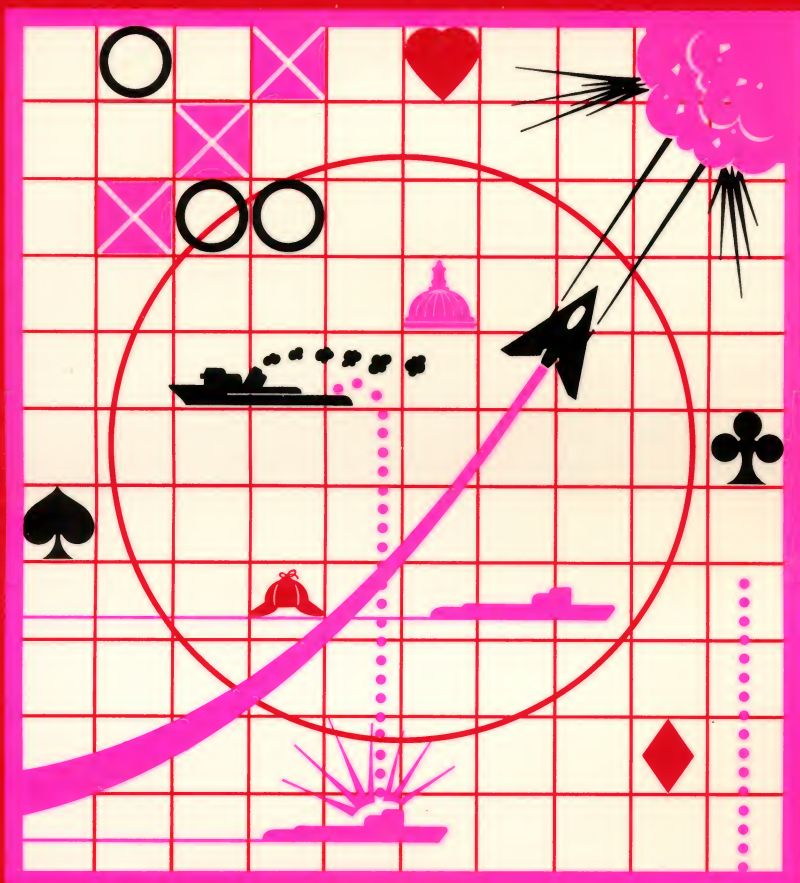


Apple® Games

Allen L. Wyatt



For Apple® II Compatible Systems

LIMITED SOFTWARE WARRANTY

This warranty applies only to the software portion of this product. If you purchased this book by itself, without the companion diskette or cassette tape, then this section does not apply to you.

For a period of ninety (90) days from the date of original purchase at retail, the warrantor, identified below, warrants this software to load and run as a basic program for the indicated microcomputer model, to be free from defects in material and workmanship and to be merchantable and suitable for its stated purpose for the period of this warranty. This warranty may not be enlarged except in writing, signed by warrantor. THE WARRANTOR EXPRESSLY DISCLAIMS ANY IMPLIED WARRANTY INCLUDING THE WARRANTY OF MERCHANTABILITY AND THE WARRANTY THAT THE SOFTWARE IS SUITABLE FOR ITS STATED PURPOSE AS OF THE DATE NINETY (90) DAYS FROM THE ORIGINAL PURCHASE OF THE SOFTWARE AT RETAIL.

In the event of defect, malfunction or failure of the software to conform with this warranty, the warrantor will repair or replace the software at no cost to you. For warranty service, you should return the software to the warrantor, **Howard W. Sams & Co., Inc., Attn: Sams Software, 4300 W. 62nd Street, Indianapolis, Indiana 46268.** Software received damaged as a result of shipping will require you to file a claim with the carrier. This warranty gives you specific legal rights and you may also have some other rights which vary from state to state.

THIS WARRANTY IS LIMITED SOLELY TO THE ABOVE AND THIS WARRANTY AND ANY WARRANTIES IMPLIED BY STATE LAW WILL APPLY ONLY FOR THE PERIOD SET FORTH. (SOME STATES DO NOT ALLOW LIMITATION ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.) THE WARRANTOR WILL NOT BE LIABLE FOR ANY LOSS, DAMAGE, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND, WHETHER BASED UPON WARRANTY CONTRACT OR NEGLIGENCE, AND ARISING IN CONNECTION WITH THE SALE, USE OR REPAIR OF THE SOFTWARE. (SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.) UNLESS OTHERWISE CONTRARY TO STATE LAW GOVERNING THE PURCHASE, THE WARRANTOR'S LIABILITY SHALL NOT IN ANY CASE EXCEED THE CONTRACT PRICE FOR THE SOFTWARE CLAIMED TO BE DEFECTIVE OR UNSUITABLE.

WARNING: THE UNAUTHORIZED USE, REPRODUCTION OR DUPLICATION OF THIS MATERIAL, OR ITS PUBLIC PERFORMANCE OR DISPLAY, BY ANY MEANS IN ANY MEDIA FOR ANY PURPOSE, WHETHER IN WHOLE OR IN PART, IS STRICTLY PROHIBITED. VIOLATORS WILL BE SUBJECT TO ALL CIVIL AND CRIMINAL PENALTIES.

Apple® Games

Apple II



Apple II Games



Allen Wyatt has been actively involved with the microcomputer industry for six years and is currently software development supervisor for Sams Software in Indianapolis, Indiana. Mr. Wyatt has had extensive experience in computer consulting and software development.

He has written several commercial software packages utilizing many of the same techniques detailed in *Apple Games*. The broad range of computer programs runs the gamut from small system data bases to games and utilities.

In addition to being a computer author, Allen is a devoted family man and active church member. He uses his personal computers to assist him in all of these areas. At home, his family spends many hours using the computer every day.

Apple[®] Games

by
Allen Wyatt

Howard W. Sams & Co., Inc.
4300 WEST 62ND ST. INDIANAPOLIS, INDIANA 46268 USA

Copyright © 1984 by Allen Wyatt

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this book, the publisher assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

International Standard Book Number: 0-672-22394-5
Library of Congress Catalog Card Number: 84-51166

Edited by *Susan Pink*

Printed in the United States of America.

Apple is a registered trademark of Apple Computer Inc.

Preface

This book is an expression of the simple things in life. We often face challenges, particularly in technological areas, that are so complex they only “boggle the mind.” *Apple Games* goes beyond that or short of that, depending on your perspective. It is short, simple, and sweet. This book shows you how several complete, ready-to-run, and entertaining games were conceived, designed, and implemented.

In addition to game programs, there are two utility programs. *Shape Table Generator* can be used to develop your own Apple graphics. *Master Catalog* can be used to bring order to your collection of programs.

Apple Games contains intermediate-level programs. Most are written in BASIC, although some include special machine-language routines that increase program speed and execution. There are complete listings and explanations for every program.

Now, your question might be “What can I do with these games?”. Well, you could play them, but that is not their greatest value. Because the concepts used in these games are documented and explained, they can be used in your programming. In this way, you will not only have fun, you will learn.

Take time to study the program listings. Then, if you feel adventurous, “poke around” and change lines to see “what happens.” You may be surprised at what you can do.

One of the easiest ways to test your new programming skills is to change a program so it uses different input. For example, some of the programs use game paddles and/or joysticks. If you don’t have either, change the input routines to use the keyboard. By changing appropriate sections, you can learn more about programming.

If you purchased this book as part of a Combo Pack, you have the programs on disk. If you bought the book separately, you have to type in the listings. The programs, when entered as they appear in the book, are designed to be controlled from a main menu. The menu program is in Section 1. It is best to begin reading from there.

NOTE: The first line of each program includes a REMark statement. This statement contains the name of the program. Use this name when saving the program to disk, otherwise, the menu will not work correctly.

Each chapter is comprised of several parts. The first part is a general statement about the program. The second is the rules or instructions for using the game or program. The third is a set of programming notes that will explain the operation of different parts of the program. Finally, each chapter contains the program listing for the game or program detailed.

As you play the games, study the programs, and use, explore and learn from this book, I hope you will have more than a little fun. After all, that is part of what computers and games are all about.

Allen Wyatt

*This book is dedicated to my children, Allen Lee and Eric Christopher.
May they always discover the joy of learning through play.*

Contents

Section 1 — THE MAIN MENU PROGRAM

THE MAIN MENU PROGRAM	12
-----------------------------	----

Section 2 — GAME PROGRAMS

1. FLIP FLOP	16
2. MASTERMIND	20
3. TOWERS	23
4. SHERLOCK'S HOME	26
5. ATTACK OF THE ZARGONS	35
6. PHASER PRACTICE	39
7. ACEY-DUCEY	43
8. BIG GOVERNMENT	49
9. TIC-TAC-TOE	57
10. QUBIC	61
11. DEPTH CHARGE	66

Section 3 — UTILITY AND MISCELLANEOUS PROGRAMS

12. SHAPE TABLE GENERATOR	72
13. OPENING CEREMONIES	78
14. MASTER CATALOG	83

A NOTE TO THE READER

The programs in this book were not written as applications software but as educational examples of what your personal computer can do. All of the programs have been tested and work on the machine configuration for which they were designed. The programs are unprotected. This means that you can modify them to better understand how they work or to fit a different machine configuration.

What Is a Combo Pack?

A Combo Pack, like this package, is a step beyond your average technical book. While most books give you programming examples through printed listings (which we do here), Combo Packs provide the book and the listings recorded on magnetic media, either disk, cassette tape, or both.

Every effort has been made to be clear, concise, and informative about how these programs and routines work. If you experience any difficulty with the software operations, the solution can be found in the book or in your computer manuals.

We are rather proud of the time and effort that went into preparing the Combo Pack. If you have purchased the Combo Pack and have enjoyed using it, let us know your thoughts. Your comments will be valuable in preparing future Combo Packs.

LOADING INSTRUCTIONS

The disk accompanying this Combo Pack contains the program listings printed in the book. To use the *Apple Games* disk, you must have a 48K Apple II compatible system, with Applesoft® in ROM or on a Language (RAM) card, and one disk drive with DOS 3.3. Game paddles or a joystick are required on selected games.

To use *Apple Games*, follow these steps:

1. With your computer turned off, insert the *Apple Games* disk in the disk drive (drive one if you have more than one disk drive).
2. Turn your computer on.
3. In a few moments, the *Apple Games* demonstration screen will be displayed.
4. Press <ESC> to exit the demonstration screen.

5. The Main Menu will be displayed. Press <RETURN> to see the second screen of menu choices. Follow the instructions in Section 1 (The Main Menu Program).

The *Apple Games* disk is write protected; consequently, you cannot make changes to the disk. To alter any of the programs (and as a precautionary measure) make a backup copy of the disk as follows:

1. With your computer turned off, insert a DOS 3.3 System Master (supplied by Apple) in the disk drive (drive one if you have more than one disk drive).
2. Turn your computer on.
3. In a few moments, you will see the Apple prompt.
4. Type **COPYA** and follow the directions on the screen.
5. Label the backup *Apple Games* disk and put the original in a safe place.



Section 1

THE MAIN MENU PROGRAM

The Main Menu Program

Most disk-based programs are menu driven; the available programs or functions can be selected from a list of choices that are clearly and conveniently displayed on the screen. *Master Catalog* (Program 14) uses menus extensively.

The menu for the *Apple Games* system is simple. It presents 15 choices on two screens. The games are listed on the first screen and the remaining (miscellaneous) programs are on the second.

In addition to the program choices, the menu also has an option to exit the program. It is a good idea to include a way to exit, because the user could become confused and want to end the program, or he might want to go on to something else.

Program Instructions

1. The program runs automatically after booting the *Apple Games* disk (after you press <ESC> to exit the demonstration screen).
2. Press <RETURN> to see the other screen of menu choices.
3. Enter the number that corresponds to the program you want to run.
4. Press <RETURN>. The disk light will come on as the program is loading, then the program will begin.
5. When you exit the menu, the computer returns to the Applesoft BASIC command level.

Programming Notes

The menu program should be saved under the name MENU so when you exit a program you can return to the menu to choose another.

10-40	Program initialization
50-110	Print first page of menu choices
120	Wait for your input
130-150	Print second page of menu choices
160	Wait for your input
170	Loop to first-page display
180-200	Prompt and receive your input
210-260	Run selected program
1000-1050	Program name data

Main Menu.

```
10      HGR :TEXT :HOME
20      NP = 15:G = 11:M = 15
30      DIM P$(NP)
40      FOR X = 1 TO NP:READ P$(X):NEXT
50      HOME :PRINT TAB( 14);"APPLE GAMES"
60      VTAB 20:POKE 34,3:HOME
70      PRINT "GAME PROGRAMS:"
80      PRINT :PRINT
90      FOR X = 1 TO G:IF X < 10 THEN PRINT " ";
100     PRINT X;" " ";P$(X)
110     NEXT
120     GOSUB 180
130     HOME :PRINT "MISCELLANEOUS:":PRINT :PRINT
140     FOR X = G + 1 TO M:PRINT X;" " ";P$(X)
150     NEXT
160     GOSUB 180
170     HOME : GOTO 70
180     VTAB 21:INPUT "CHOICE OR <RETURN> FOR MORE: ";C
190     H$
200     IF CH$ = CHR$( 13) THEN RETURN
210     CH = VAL (CH$):IF CH < 1 OR CH > NP THEN RETURN
220     POKE 34,0:PRINT
230     VTAB 21:HTAB 1:CALL - 868
240     IF CH = NP THEN TEXT :HOME :PRINT "HAVE A NICE D
AY!": GOTO 260
250     A$ = "NOW LOADING " + P$(CH):PRINT TAB( 20 - LEN
(A$) / 2);A$
260     PRINT "RUN";P$(CH)
270     END
1000    DATA FLIP FLOP,MASTERMIND,TOWERS,SHERLOCK'S HOME
1010    DATA ACEY-DUCEY,BIG GOVERNMENT
1020    DATA TIC-TAC-TOE,QUBIC,DEPTH CHARGE
1030    DATA SHAPE TABLE GENERATOR
1040    DATA OPENING CEREMONIES,MASTER CATALOG
1050    DATA EXIT PROGRAMS
```



Section 2

GAME PROGRAMS

1. Flip Flop

This is an easy-to-play game of chance. You guess what the computer's next move will be. As you will soon discover, this can be very aggravating.

The object of the game is to change a row of 10 X's to 10 O's in as few moves as possible. You do this by choosing the position of the character you want to change. Sounds easy, right? Well, there's a catch—the computer also may randomly change a character. The computer could change the position you selected in the previous move, thus negating your choice. Take heart, however, because the odds are in your favor.

You may find *Flip Flop* not only aggravating, it may also be addictive. Your chance of winning depends on whether or not the Powers-That-Be recognize the sincerity of your efforts. Look sincere and don't berate the computer if it takes you longer to win than you first thought. After all, your computer is only a tool for the Powers-That-Be!

Game Rules

1. If you are selecting the game from the *Main Menu*, choose 1. If you are not using the menu, run the program.
2. The directions will be displayed.
3. Press a 0 through 9 to change the corresponding X to an O, or O to X.
4. Try to change all 10 X's to O's.
5. Pressing <ESC> at any time ends the game.
6. When the game is finished, you can play again or return to the Main Menu.

Programming Notes

This game utilizes text and sound. One of the first steps is to POKE a very short machine-language sound-generation routine into page 3 of memory. This routine is then called throughout the program when sound is desired. Fig. 1-1 shows a "disassembled" listing of the routine; because it is used in several programs, you may want to study it.

Ø3ØØ-	A6 Ø1		LDX \$Ø1	* Duration
Ø3Ø2-	A4 ØØ	LOOP1	LDY \$ØØ	* Pitch
Ø3Ø4-	AD 3Ø CØ		LDA \$CØ3Ø	* Toggle speaker
Ø3Ø7-	88	LOOP2	DEY	
Ø3Ø8-	DØ FD		BNE \$Ø3Ø7	* To Loop2
Ø3ØA-	CA		DEX	
Ø3ØB-	DØ F5		BNE \$Ø3Ø2	* To Loop1
Ø3ØD-	6Ø		RTS	* Finished

Fig. 1-1. Sound-generation routine.

First, the pitch and duration has to be set when the routine is called. This is done by storing (using the POKE command) the desired pitch in memory location Ø, and the duration in memory location 1. Then, a call is made to 768 (\$3ØØ) to produce the sound.

Although this routine is short, it is all that is needed to produce a controlled sound. The sound is produced by toggling (or clicking) the speaker. However, if it was produced using one click right after another you would only hear a buzz. To eliminate the buzz, there is a short delay between each click on the speaker. This is what produces the pitch. Using different pitch values in memory location Ø changes the time delay between each click of the speaker, thus producing different tones.

The duration is the length of time the tone will play. The larger the number stored in memory location 1, the longer each tone. In this program, we produce a short tone by using a value of 2Ø.

The sound-generation routine is used in the subroutine in lines 54Ø through 56Ø. Each character of Z\$ is printed, one at a time, followed by a short, randomly chosen, musical note. A "trilling" effect is produced as each character in the string is displayed.

1Ø-2Ø	Program initialization and sound routine
3Ø-18Ø	Instructions
19Ø-2ØØ	Game initialization
21Ø-22Ø	Get string and print it
23Ø-4ØØ	Get your input and randomly change X's to O's or O's to X's
41Ø	Increment turn counter
42Ø-43Ø	Check for winner and loop if none
44Ø-45Ø	Delay and print final string value
46Ø-48Ø	End of game message
49Ø-52Ø	Play again or exit to Main Menu

53Ø Concatenate game array to a string
 54Ø-56Ø Print a string while playing random sounds

Flip Flop.

```

1Ø TEXT :HOME
2Ø POKE 768,166:POKE 769,1:POKE 77Ø,164:POKE 771,Ø:
POKE 772,173:POKE 773,48:POKE 774,192:POKE 775,1
36:POKE 776,2Ø8:POKE 777,253:POKE 778,2Ø2:POKE 7
79,2Ø8:POKE 78Ø,245:POKE 781,96
3Ø SPEED= 175
4Ø Z$ = "THE OBJECT OF THE ANCIENT GAME OF":VTAB 2:
GOSUB 54Ø
5Ø Z$ = "FLIP FLOP":GOSUB 54Ø
6Ø Z$ = "IS TO CHANGE THIS":GOSUB 54Ø
7Ø Z$ = "X X X X X X X X X X":VTAB 6:GOSUB 54Ø
8Ø Z$ = "TO THIS":VTAB 8:GOSUB 54Ø
9Ø Z$ = "O O O O O O O O O":VTAB 1Ø:GOSUB 54Ø
1ØØ Z$ = "IN AS FEW MOVES AS POSSIBLE.":VTAB 12:GOSU
B 54Ø
11Ø Z$ = "TO CHANGE A POSITION, PRESS THE NUMBER":VT
AB 14:GOSUB 55Ø
12Ø Z$ = "OF THAT POSITION. IT WILL THEN BE":GOSUB
55Ø
13Ø Z$ = "CHANGED. HOWEVER, THE GODS MAY NOT":GOSUB
55Ø
14Ø Z$ = "SMILE GRACIOUSLY UPON YOUR EFFORTS, AND":G
OSUB 55Ø
15Ø Z$ = "THE POSITIONS MAY CHANGE ON THEIR OWN.":GO
SUB 55Ø
16Ø Z$ = "...GOOD LUCK !":VTAB 2Ø:GOSUB 55Ø
17Ø Z$ = "P.S.: TO QUIT THE GAME, PRESS <ESC>." :VTAB
22:GOSUB 55Ø
18Ø SPEED= 255:FOR X = 1 TO 3ØØØ:NEXT :HOME
19Ø FOR X = 1 TO 1Ø:A$(X) = "X":NEXT
2ØØ VTAB 8:HTAB 1Ø:PRINT "1 2 3 4 5 6 7 8 9 Ø"
21Ø GOSUB 53Ø
22Ø VTAB 9:HTAB 1:CALL - 868:FOR X = 1 TO 3ØØ:NEXT
:HTAB 1Ø:GOSUB 55Ø
23Ø VTAB 2Ø:PRINT "TO CHANGE: ";
24Ø GET CH$:IF CH$ = CHR$( 27) THEN HOME :PRINT : GO
TO 49Ø
25Ø IF ASC (CH$) < 48 OR ASC (CH$) > 57 THEN 24Ø
26Ø N = VAL (CH$):IF N = Ø THEN N = 1Ø
27Ø PRINT :M = N
28Ø IF A$(N) = "O" THEN 32Ø
29Ø A$(N) = "O"
3ØØ N = INT (RND (1) * 11):IF N < 1 OR N > 1Ø THEN 3
ØØ
31Ø IF A$(N) = "X" THEN A$(N) = "O": GOTO 41Ø
32Ø A$(N) = "X"
33Ø IF M = N THEN 3ØØ
34Ø GOTO 41Ø
35Ø IF A$(N) = "O" THEN 39Ø

```

```

360  A$(N) = "O"
370  N = INT (RND (1) * 11):IF N < 1 OR N > 10 THEN 3
    70
380  IF A$(N) = "X" THEN A$(N) = "O": GOTO 410
390  A$(N) = "X"
400  IF M = N THEN 370
410  C = C + 1
420  FOR X = 1 TO 10:IF A$(X) < > "O" THEN 210
430  NEXT
440  FOR X = 1 TO 300:NEXT
450  GOSUB 530:VTAB 9:HTAB 1:CALL - 868:HTAB 10:GOSU
    B 550
460  VTAB 20:HTAB 1:CALL - 868
470  IF C < 13 THEN PRINT "VERY GOOD. YOU GUESSED IT
    IN ONLY ";C:PRINT "GUESSES !!!": GOTO 490
480  PRINT "GOOD TRY...BUT YOU COULD DO BETTER.":PRIN
    T "IT TOOK YOU ";C;" GUESSES THIS TIME."
490  PRINT :PRINT "TRY AGAIN (Y/N): ";
500  GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 50
    0
510  IF CH$ = "Y" THEN HOME :C = 0: GOTO 190
520  HOME :VTAB 8:HTAB 10:PRINT "THANX FOR THE GAME !
    !":VTAB 1:HTAB 1:PRINT "RUN MENU"
530  Z$ = "":FOR X = 1 TO 10:Z$ = Z$ + A$(X) + " ":NE
    XT :RETURN
540  POKE 36,20 - LEN (Z$) / 2
550  FOR X = 1 TO LEN (Z$):PRINT MID$( Z$,X,1);:IF MI
    D$( Z$,X,1) < > " " THEN POKE 0,RND (1) * 80 +
    50:POKE 1,20:CALL 768
560  NEXT :PRINT :RETURN

```

2. Mastermind

Most people are familiar with the game of *Mastermind*, right? Well, this version has a new "twist," because that is what you do! You twist the game paddle (paddle "Ø") to change the colors and use the paddle button to "lock in" the color of your choice. Joysticks can be used instead of game paddles.

This game illustrates the use of low-resolution graphics and game paddles or joysticks. No text is displayed. When you finish guessing the color and position of a row of squares, the result is displayed with colored dots (see game rule #5) instead of a message. Thus, this game is well-suited for small children who have not learned to read. However, it is also a refreshing change of pace for older children and adults.

Mastermind is designed to be used with a color display. If you have a monochrome monitor, it is difficult, although not impossible, to play. The colors of the squares and the results will appear as different patterns instead of different colors (see the Game Rules). The type of patterns displayed depends on the type of monitor you have.

Game Rules

1. If you are selecting the game from the Main Menu, choose 2. If you are not using the menu, run the program.
2. The directions will be displayed.
3. Turn the game paddles or move the joystick controller until the desired color is displayed at the blinking rectangle.
4. Press the button on the game paddle or joystick. The next position will begin blinking, or, if you have completed a row, the results will be displayed.
5. For every square where you guessed the correct color, but the wrong position, a turquoise dot is displayed. If you guessed the correct color and the correct position, a pink dot is displayed.
6. When you have guessed the colors for the first row, try again on the next row.
7. To end the game at any time, press <ESC>.

8. You win when you have matched the position and color of every square in the row.
9. When completed, the program returns to the Main Menu.

Programming Notes

10-95	Instructions
100-110	Game initialization
120-140	Display input line and get your input
150-210	Check for correct input
220-240	Display results
250-260	Check for winner and loop if no winner yet
270-290	End the game
300-350	Display input line
360-510	Get input from paddles or <ESC> to end game
520-570	Randomly select the game elements
580	Blank out graphics line (the row of squares)

Mastermind.

```

10 REM MASTERMIND
20 SPEED= 100
30 HOME :PRINT TAB( 15);"MASTERMIND":PRINT
40 PRINT "THE OBJECT OF MASTERMIND IS TO GUESS THE CORRECT COLOR COMBINATION OF FOUR"
50 PRINT "SQUARES. THE COMPUTER WILL ALLOW YOU TO CHANGE THE COLOR OF EACH POSITION BY USE OF THE GAME PADDLES. WHEN THE COLOR YOU WISH APPEARS, PRESS THE PADDLE BUTTON,"
60 PRINT "AND YOU CAN CHANGE THE NEXT SQUARE.":PRINT "WHEN ALL FOUR SQUARES ARE COMPLETED,":PRINT "YOU WILL SEE HOW MANY SQUARES WERE THE"
70 PRINT "RIGHT COLOR, AND IF THEY WERE IN THE":PRINT "RIGHT POSITION.":PRINT
80 PRINT "FOR EVERY ONE THAT WAS A CORRECT COLOR,":PRINT "BUT IN THE WRONG POSITION, YOU WILL SEE":PRINT "A SMALL TURQUOISE DOT, AND FOR EVERY"
90 PRINT "ONE THAT WAS THE CORRECT COLOR AND ALSO":PRINT "IN THE CORRECT POSITION, YOU WILL SEE A":PRINT "SMALL PINK DOT."
95 PRINT :PRINT "PRESSING <ESC> AT ANY TIME ENDS THE":PRINT "GAME."
100 GOSUB 520:SPEED= 255:HOME
110 Q = - 4:T = 0:GR
120 Q = Q + 5:IF Q > 37 THEN Q = 1
130 H = 0:T = T + 1
140 GOSUB 580:GOSUB 300:GOSUB 360
150 RW = 0:RR = 0
160 FOR X = 1 TO 4:F(X) = 0
170 IF G(X) = M(X) THEN RR = RR + 1:F(X) = 1

```

```

180     NEXT
190     FOR X = 1 TO 4:FOR Y = 1 TO 4
200     IF G(Y) = M(X) AND Y < > X AND F(X) = 0 THEN RW
        = RW + 1:F(X) = 1
210     NEXT :NEXT
220     H = 17
230     IF RW > 0 THEN COLOR= 14:FOR X = 1 TO RW:H = H +
        3:PLOT H,Q:NEXT
240     IF RR > 0 THEN COLOR= 11:FOR X = 1 TO RR:H = H +
        3:PLOT H,Q:NEXT
250     IF RR < 4 THEN 120
260     PRINT "CONGRATULATIONS, YOU DID IT IN ";T:PRINT
        "TURNS!":PRINT
270     VTAB 23:PRINT "PRESS <RETURN> TO CONTINUE...";
280     GET CH$:IF CH$ < > CHR$(13) THEN 280
290     TEXT :HOME :PRINT :PRINT CHR$(4)"RUN MENU"
300     COLOR= 10
310     HLIN 0,3AT Q:HLIN 0,3AT Q + 1
320     HLIN 5,8AT Q:HLIN 5,8AT Q + 1
330     HLIN 10,13AT Q:HLIN 10,13AT Q + 1
340     HLIN 15,18AT Q:HLIN 15,18AT Q + 1
350     RETURN
360     FOR X = 1 TO 4:G(X) = 0
362     CH = PEEK ( - 16384):IF CH < 128 THEN 370
364     POKE - 16368,0:IF CH = 155 THEN POP : GOTO 270
370     COLOR= 0
380     HLIN H,H + 3AT Q:HLIN H,H + 3AT Q + 1
390     FOR Z = 1 TO 100:NEXT
400     C = INT (PDL (0) / 17)
410     IF C < 1 THEN C = 1
420     IF C > 3 AND C < 7 THEN C = 4
430     IF C < 11 AND C > 6 THEN C = 9
440     IF C > 10 AND C < 14 THEN C = 13
450     IF C > 13 THEN C = 15
460     COLOR= C
470     HLIN H,H + 3AT Q:HLIN H,H + 3AT Q + 1
480     FOR Z = 1 TO 100:NEXT
490     IF PEEK ( - 16287) < 128 THEN 362
500     G(X) = C
510     H = H + 5:NEXT X:RETURN
520     FOR X = 1 TO 4:M(X) = 0
530     M = INT (RND (2) * 15):IF M > 13 THEN 530
540     IF M = 0 OR (M > 4 AND M < 9) OR (M > 9 AND M <
        13) THEN 530
550     FOR Y = 1 TO X:IF M = M(Y) THEN 530
560     NEXT Y
570     M(X) = M:NEXT X:RETURN
580     COLOR= 0:HLIN 0,30AT Q:HLIN 0,30AT Q + 1:RETURN

```


3. Towers

You've heard of the famous Leaning Tower of Pisa? That's good, even though it has nothing to do with this game.

The proper name for this game is *The Towers of Hanoi*. It begins with several disks of different sizes, located on the leftmost of three towers. The object of the game is to move all the disks to tower three, the rightmost tower, in as few moves as possible. However, you can only move one disk at a time and you cannot place a larger disk on a smaller one. Try playing with two disks and then progress to three.

If you take too many moves, you lose the game. Besides that, you may not get invited back to Hanoi.

This game demonstrates screen control techniques. The needles and disks appear to remain static, while the comments and prompts scroll on the bottom portion of the screen. This is done using the Apple window-control memory locations, and can be examined in the display routine in lines 500 through 530.

Game Rules

1. If you are selecting the game from the Main Menu, choose 3. If you are not using the menu, run the program.
2. The directions will be displayed.
3. Choose the number of disks to use in the game. You may select one through seven disks.
4. Press <RETURN> to begin.
5. Select a disk to move. Disks are represented with 3, 5, 7, 9, 11, 13, or 15 asterisks (*).
6. You can only move the top disk on a tower (needle).
7. Select which tower (needle) to move the disk to. Needles are specified by 1, 2, and 3, 1 is the leftmost needle.
8. You cannot place a large disk on a smaller one.
9. Try to get all the disks to needle 3, the rightmost one, in as few moves as possible.
10. When completed, you can play again or exit to the Main Menu.

Programming Notes

10-60	Initialization and instructions
70-90	Determine number of disks to use by prompting player
100-150	Additional instructions
160-170	Prompt for <RETURN> to be pressed
180-200	Game initialization
210-300	Determine disk to move and check if valid move
310-400	Determine needle to move to and check if valid move
410	Refresh screen (by redrawing the display)
420	Increment move counter and stop if too many moves
430-440	Check if winner, loop if none yet
450	Print game move results
460-490	Prompt for new game or exit to Main Menu
500-530	Display needles and disks on screen
540	Exit game

Towers.

```
10      REM      TOWERS
20      DIM F(3,7)
30      FOR Q = 1 TO 3:FOR R = 1 TO 7:F(Q,R) = 0:NEXT :H
      OME :PRINT :PRINT
40      PRINT TAB( 9);"TOWERS OF HANOI PUZZLE":PRINT
50      PRINT "YOU MUST TRANSFER THE DISKS FROM THE L
      EFT TO THE RIGHT TOWER, ONE AT A TIME,"
60      PRINT "NEVER PUTTING A LARGER DISK ON A":PRINT "
      SMALLER ONE.":PRINT
70      PRINT "HOW MANY DISKS (MAXIMUM OF 7): ";
80      GET S$:S = VAL (S$):IF S < 1 OR S > 7 THEN 80
90      PRINT S:PRINT :PRINT :M = 0
100     HOME :PRINT "IN THIS GAME, WE WILL REFER TO DISK
      S BY":PRINT "A NUMERICAL CODE.":PRINT
110     PRINT "3 WILL REPRESENT THE SMALLEST DISK, 5 T
      HE NEXT SIZE, 7 THE NEXT, AND SO ON, "
120     PRINT "UP TO 15. YOU CAN IDENTIFY A DISK BY":PR
      INT "THE NUMBER OF ASTERISKS WITH WHICH IT":PRIN
      T "IS MADE.":PRINT
130     PRINT "THE NEEDLES ARE NUMBERED FROM LEFT TO":PR
      INT "RIGHT, 1 TO 3. WE WILL START WITH THE":PRI
      NT "DISKS ON NEEDLE 1, AND ATTEMPT TO MOVE":PRIN
      T "THEM TO NEEDLE 3."
140     PRINT :PRINT "GOOD LUCK!!!!":PRINT
150     PRINT
160     PRINT "PRESS <RETURN> TO BEGIN...";
170     GET A$:IF A$ < > CHR$( 13) THEN 170
180     HOME
190     Y = 7:FOR Q = S TO 1 STEP - 1:F(1,Y) = Q * 2 +
      1:Y = Y - 1:NEXT
200     GOSUB 500
```

```

210 PRINT "WHICH DISK WOULD YOU LIKE TO MOVE: ";
220 D = 0
230 GET S$:IF S$ = CHR$(8) THEN PRINT : GOTO 210
235 D = D * 10 + VAL (S$):IF VAL (S$) = 0 THEN 230
240 IF D = 1 THEN PRINT D;: GOTO 230
250 PRINT VAL (S$)
260 IF ((D - 1) / 2) > 0 AND ((D - 1) / 2) < = S TH
EN 290
270 PRINT "YOU MAY ONLY TYPE ";:FOR J = 1 TO S:PRINT
J * 2 + 1;:IF J < > S THEN PRINT ", ";
280 NEXT :PRINT : GOTO 210
290 FOR Q = 1 TO 3:FOR R = 1 TO 7:IF F(Q,R) < > D T
HEN NEXT :NEXT :PRINT "SORRY, BUT THAT DISK IS N
OT BEING USED IN THIS GAME.": GOTO 210
300 IF F(Q,R - 1) > 0 THEN PRINT "THAT DISK IS BELOW
ANOTHER ONE. MAKE":PRINT "ANOTHER CHOICE.": GO
TO 210
310 PRINT "PLACE DISK ON WHICH NEEDLE (1-3): ";
320 GET S$:N = VAL (S$):IF N < 1 OR N > 3 THEN 320
330 PRINT N
340 FOR Q = 1 TO 7:IF F(N,Q) = 0 THEN NEXT : GOTO 36
0
350 IF D > = F(N,Q) THEN PRINT "SORRY, BUT YOU CAN'
T MAKE THAT MOVE!": GOTO 210
360 FOR U = 1 TO 7:IF F(N,U) = 0 THEN NEXT U
370 U = U - 1
380 FOR Q = 1 TO 3:IF Q = N THEN 400
390 IF F(Q,R) = D THEN F(N,U) = F(Q,R):F(Q,R) = 0:Q
= 3
400 NEXT Q
410 GOSUB 500
420 M = M + 1:IF M > 128 THEN PRINT "SORRY, BUT I HA
VE ORDERS TO STOP IF YOU MAKE MORE THAN 128 MOVE
S.": GOTO 540
430 FOR Q = 1 TO 2:FOR R = 1 TO 7:IF F(Q,R) < > 0 T
HEN 210
440 NEXT :NEXT
450 POKE 34,0:HOME :VTAB 8:PRINT "YOU HAVE PERFORMED
THE TASK IN ";M;" MOVES":PRINT
460 PRINT "WOULD YOU LIKE ANOTHER GAME (Y/N): ";
470 GET A$:IF A$ < > "Y" AND A$ < > "N" THEN 470
480 PRINT A$:IF A$ = "N" THEN 540
490 CLEAR : GOTO 30
500 POKE 34,0:POKE 35,22:VTAB 1:HTAB 1
510 FOR Q = 1 TO 3:Z = Q * 10:FOR R = 1 TO 7:VTAB (Q
- 1) * 7 + R:HTAB 1:CALL - 868
520 HTAB (Z - INT (F(Q,R) / 2)):FOR Y = 1 TO F(Q,R):
PRINT "*";:NEXT :IF R = 7 THEN HTAB Z:INVERSE :P
RINT Q;:NORMAL
530 NEXT :NEXT :POKE 35,24:POKE 34,21:VTAB 24:HTAB 1
:RETURN
540 TEXT :PRINT :PRINT CHR$(4)"RUN MENU"

```

4. Sherlock's Home

Elementary, my dear Watson! In this game, you're the master detective, and you have to discover "who dunnit" before he does you in. *Sherlock's Home* uses one- and two-word commands, similar to many adventure or fantasy games.

Each time you are called in on a new case, there is a different victim. But the suspects remain the same. Not surprising to a trained detective like yourself, is it?

The only fact the police have turned up is that the suspect resides in Sherlock's home. But you have a few hunches yourself, and one of them may prove to be correct — he's out to get you too. You can move from room to room, pick up weapons, and make announcements as to who you think committed the dire deed. If any of your guesses (as to the suspect, room, or weapon) are wrong, you get the reason why, but nothing else. And time isn't on your side — each step you take brings your nemesis closer and closer.

Game Rules

1. If you are selecting the game from the Main Menu, choose 4. If you are not using the menu, run the program.
2. The instructions will be displayed.
3. Move from room to room to determine who committed the crime and how it was done.
4. When proposing a solution, you must be in the room where you think the crime was committed.
5. You must be holding the correct weapon to win.
6. You must deduce who committed the crime, where it happened, and what weapon was used.
7. You can move from room to room by typing **GO** followed by the name of the room. The rooms are:

LIVING ROOM
STUDY
BEDROOM
BASEMENT

DEN
KITCHEN
ATTIC

DINING ROOM
WATERCLOSET
GUEST ROOM

8. If a weapon is in the room, you can pick it up by typing **TAKE** followed by its name, but you can carry only one. Possible weapons are:

GUN
KNIFE

ROPE
POISON

CANDLESTICK
WRENCH

9. To drop a weapon, type **DROP** followed by its name.
10. The suspects and their occupations are:
- | | |
|---------------|----------------------|
| MR. MISER | Millionaire |
| MRS. SIPPY | Housewife |
| CISSI SIPPY | Housewife's daughter |
| COL. KENTUCKY | Restaurant magnate |
| OLIVE PITTS | Fruit grower |
| HEAVES | Family butler |
| SPOT | Family dog |
11. If you think you have solved the case, make an announcement by entering an asterisk (*) as your command.
12. The game ends when you get the criminal or he gets you.
13. Upon completion, you can play again or exit to the Main Menu.

Programming Notes

10-290	Initialization
300-690	Display instructions
700-715	Prompt for displaying the instructions again
720-830	Game initialization
840-930	Main program loop
870	Prompt for command
880	Check detective's announcement
890	Check command to go to another room
900	Check command to take a weapon
910	Check command to drop a weapon
940-960	You're too late, you've been done in
970-980	Announce "who dunnit"
990-1320	Routine for detective's announcement
990	Check for no weapon in hand
1000-1070	Display announcement screen

	1080-1090	Get your guess of who dunnit
	1100-1130	Check theory in order of room, weapon, and person
	1140	You solved it!
	1150-1180	Play again or exit to the Main Menu
	1190-1200	Sorry, wrong room
	1210-1220	Sorry, wrong weapon
	1230-1240	Sorry, wrong person
	1250-1280	Check theory in order of weapon, person, room
	1290-1320	Check theory in order of person, room, weapon
1330-1460		Routine for going to a room
	1330-1390	Check for valid room
	1400	In that room already
	1410-1440	Special routine for watercloset
	1450-1460	Change room
1470-1640		Routine for taking a weapon
	1470-1520	Check for valid weapon
	1530	Valid weapon, but not in room
	1540-1560	Check for weapon in room
	1570	Weapon not in room
	1580-1640	Transfer weapon from in the room to in the detective's possession
1650-1740		Routine for dropping a weapon
	1660	Sorry, not carrying anything to drop
	1670-1710	Check for valid weapon
	1720	Not carrying that weapon
	1730-1740	Transfer weapon from in the detective's possession to in the room
1760-1770		Get random number between 1 and 10
1780-1800		Press <RETURN> to continue
1810-1890		Display information (room, weapons visible, etc.)
1900-1920		End program

Sherlock's Home.

```

5      REM          SHERLOCK'S HOME
10     DATA FRED, JOHN, CARL, ZEKE, ARCHIBALD, FRANCINE, AGAT
      HA, WILLIAM, MORTIMER, FANNY
20     DATA SMITH, ABERCROMBIE, HARKENFARKER, SNERD, WHOSIT
      S, SPITZINPOT, PEEZINPEPPER, BELEAKE, GYTOPE, PEON
30     DATA LIVING ROOM, DEN, DINING ROOM, STUDY, KITCHEN, W
      ATERCLOSET, BEDROOM, ATTIC, GUEST ROOM, BASEMENT

```

```

40 DATA GUN,WRENCH,ROPE,CANDLESTICK,KNIFE,POISON
50 DATA MR. MISER,MRS. SIPPY,CISSI SIPPY,COL. KENTU
   CKY,OLIVE PITTS,HEAVES,SPOT
60 FOR X = 1 TO 10:READ F$(X):NEXT :FOR X = 1 TO 10
   :READ L$(X):NEXT
70 FOR X = 1 TO 10:READ P$(X):NEXT :FOR X = 1 TO 6:
   READ W$(X):NEXT :FOR X = 1 TO 7:READ S$(X):NEXT
80 GOSUB 1760:N$ = F$(R):F$ = N$:GOSUB 1760:N$ = N$
   + " " + L$(R)
90 DIM R(10,6)
100 V$(1) = "O.K. EVERYONE IS HERE....WHAT'S THE BIG
   ANNOUNCEMENT ?"
110 V$(2) = "LAY IT ON US, OH EXALTED DETECTIVE.....
   ."
120 V$(3) = "THINK YOU GOT THE ANSWER ? WE'LL SEE..
   ."
130 V$(4) = "I HOPE YOU ARE NOT WASTING OUR TIME....
   ."
140 V$(5) = "OH, NO.....NOT ANOTHER THEORY.....
   ."
150 V$(6) = "THE KILLER IS GETTING CLOSER....I HOPE
   YOU HAVE THE ANSWER...."
160 V$(7) = "SOMEONE IN THIS ROOM MAY KILL YOU IF YO
   UDON'T HAVE THE CORRECT ANSWER..."
170 V$(8) = "TIME IS RUNNING OUT.....DO YOU HAVE THE
   ANSWER ?"
180 V$(9) = "GETTING CLOSE, HUH ? COULD BE DANGEROU
   SAROUND HERE SOON..."
190 V$(10) = "THE SURGEON GENERAL HAS WARNED THAT TO
   O MANY THEORIES WITH THE KILLER NEARBY COULD
   BE HAZARDOUS TO YOUR HEALTH."
200 X$(1) = "NOT QUITE..."
210 X$(2) = "WRONG AGAIN..."
220 X$(3) = "ARE YOU KIDDING ?"
230 X$(4) = "WRONGO !"
240 X$(5) = "SINCE WHEN ?"
250 X$(6) = "THINK AGAIN..."
260 X$(7) = "GOOD TRY, BUT..."
270 X$(8) = "THINK SO, HUH ?"
280 X$(9) = "I DOUBT IT..."
290 X$(10) = "BAD JUDGEMENT..."
300 HOME:PRINT TAB(7);"WELCOME TO SHERLOCK'S HOME"
   :PRINT
310 PRINT " THE OBJECT OF THE GAME IS TO GUESS":PRI
   NT CHR$(34);"WHO DUNNIT";CHR$(34);". THE COMP
   UTER WILL CHOOSE"
320 PRINT "THE KILLER, WHERE THE CRIME WAS":PRINT "C
   OMMITTED, AND THE MURDER WEAPON.":PRINT "BECAUSE
   THE LOCAL POLICE ARE COMPLETELY":PRINT "BAFFLED
   BY THIS COMPUTER-ASSISTED"
330 PRINT "CRIME, YOU, AS THE WORLD'S FOREMOST":PRIN
   T "AUTHORITY ON COMPUTER CRIME (WHAT WE IN":PRIN
   T "THE TRADE REFER TO AS 'C.C. '), HAVE"
340 PRINT "BEEN CALLED IN TO DISCOVER THE ANSWER."

```

```

350 PRINT :PRINT " THE ONLY ESTABLISHED FACT IS THA
T THE CRIME TOOK PLACE SOMEWHERE IN AN OLD DE
360 PRINT "LOCK'S HOME";CHR$(34);"SHER-"
HE LATE"
370 PRINT "JOHN SPENCER SHERLOCK III, RICH AND F
AMOUS MYSTERY AUTHOR)."

```



```

        YOU THINK WAS USED TO DO THE DIRTY DEED."
640 PRINT :PRINT " YOU MAY PICK UP WEAPONS AND MOVE
        THEM FROM ROOM TO ROOM BY TYPING 'TAKE ' AND TH
        E NAME OF THE WEAPON. FOR EXAMPLE, TOPICK UP TH
        E GUN, THE COMMAND WOULD BE 'TAKE GUN'."
650 GOSUB 1780
660 PRINT " YOU CAN ONLY CARRY ONE WEAPON AT A T
        IME. YOU CAN DROP ANY WEAPON WITH THE COMMAND '
        DROP ' AND THE NAME OF THE WEAPON."
670 PRINT
680 PRINT " AS A FINAL NOTE, THE KILLER HAS AN A
        VERSION TO GETTING CAUGHT. AS TIME GOES ON,
        YOU MAY GET KILLED BY THE KILLER, SO TRY TO
        DISCOVER WHO DID THE DASTARDLY DEED BEFORE YOU
        BECOME THE NEXT VICTIM."
690 PRINT :PRINT " BY THE WAY, EVERY GOOD DETECTIVE
        HAS A LITTLE NOTEBOOK HE CARRIES AROUND WITHHI
        M. IT MIGHT BE A GOOD IDEA IF YOU DID TOO."
700 VTAB 23:PRINT "REVIEW THE INSTRUCTIONS AGAIN (Y/
        N): ";
710 GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 71
        0
715 IF CH$ = "Y" THEN 300
720 GOSUB 1760:C(1) = R
730 GOSUB 1760:IF R > 6 THEN 730
740 C(2) = R
750 GOSUB 1760:IF R > 7 THEN 750
760 C(3) = R:UL = 50:GOSUB 1760:P(1) = R:P(2) = 0:GO
        SUB 1760:UL = UL - R
770 FOR X = 1 TO 10:R(X,0) = 0:F(X) = 0:NEXT
780 FOR X = 1 TO 6
790 GOSUB 1760:IF R > 6 THEN 790
800 IF F(R) = 1 THEN 790
810 T = R:F(R) = 1
820 GOSUB 1760:R(R,R(R,0) + 1) = T:R(R,0) = R(R,0) +
        1
830 NEXT :HOME :POKE 37,10
840 FOR Z = 1 TO UL
850 GOSUB 1810
860 PRINT
870 INPUT "COMMAND ? ";I$
880 IF I$ = "*" THEN GOSUB 990: GOTO 930
890 IF LEFT$( I$,2) = "GO" THEN GOSUB 1330: GOTO 930
900 IF LEFT$( I$,4) = "TAKE" THEN GOSUB 1470: GOTO 9
        30
910 IF LEFT$( I$,4) = "DROP" THEN GOSUB 1650: GOTO 9
        30
920 PRINT :PRINT "SORRY, SIR, BUT I DON'T UNDERSTAND
        WHAT YOU MEAN BY ";I$
930 NEXT
940 TEXT :HOME :PRINT "SORRY TO REPORT,. SIR, THAT YO
        U HAVE JOINED ";N$;" IN THAT"
950 PRINT "GREAT PRECINCT HOUSE IN THE SKY.":PRINT
960 PRINT "CHALK UP ANOTHER MURDER TO ";S$(C(3));"."
        :PRINT

```

```

970 PRINT "BY THE WAY, THE MURDER WEAPON WAS THE":PRINT W$(C(2));" AND IT WAS DONE IN THE":PRINT P$(C(1));"."
980 PRINT : GOTO 1900
990 IF P(2) = 0 THEN PRINT "SORRY, BUT YOU AREN'T CARRYING A MURDER WEAPON....REMEMBER THE RULES, SIR !":PRINT :RETURN
1000 TEXT :HOME
1010 Q1 = 5:IF Z > (Z / 3) * 2 THEN Q1 = 10
1020 GOSUB 1760:IF R > Q1 THEN 1020
1030 PRINT V$(R):PRINT
1040 PRINT "ACCORDING TO YOU, THE MURDER OF POOR":PRINT N$;" WAS DONE IN THE"
1050 PRINT P$(P(1));" WITH THE ";W$(P(2));"."
1060 PRINT :PRINT "NOW, THE QUESTION IS 'WHO DID IT' ?":PRINT
1070 FOR X = 1 TO 7:PRINT X;") ";S$(X):NEXT :PRINT
1080 PRINT "WHAT IS THE CULPRIT'S NUMBER (1-7): ";
1085 GET I$:I = VAL (I$):IF I < 1 OR I > 7 THEN 1085
1090 PRINT :GOSUB 1760:IF R > 3 THEN 1090
1100 ON R GOTO 1110,1250,1290
1110 PRINT :IF C(1) < > P(1) THEN 1190
1120 IF C(2) < > P(2) THEN 1210
1130 IF C(3) < > I THEN 1230
1140 PRINT :PRINT "CONGRATULATIONS, SIR....YOU SOLVED THE CASE (PERSONALLY, SIR, I NEVER DOUBTED YOUR VERACITY FOR A MINUTE.)."
1150 PRINT :PRINT "WOULD YOU LIKE ANOTHER ASSIGNMENT, OR ARE YOU DUE FOR A VACATION (A/V): ";
1160 GET I$:IF I$ < > "A" AND I$ < > "V" THEN 1160
1165 PRINT I$
1170 IF I$ = "V" THEN TEXT :HOME :PRINT "HAVE A GOOD VACATION, SIR....HURRY BACK SOON !": GOTO 1920
1180 POP : GOTO 300
1190 PRINT :GOSUB 1760:PRINT X$(R)
1200 PRINT "THE ";P$(P(1));" WAS NOT THE CORRECT":PRINT "ROOM !":PRINT :RETURN
1210 PRINT :GOSUB 1760:PRINT X$(R)
1220 PRINT W$(P(2));" IS THE WRONG WEAPON, SIR !":PRINT :RETURN
1230 PRINT :GOSUB 1760:PRINT X$(R)
1240 PRINT S$(I);" HAS A PERFECT ALIBI,":PRINT "SIR !":PRINT :RETURN
1250 IF C(2) < > P(2) THEN 1210
1260 IF C(3) < > I THEN 1230
1270 IF C(1) < > P(1) THEN 1190
1280 GOTO 1140
1290 IF C(3) < > I THEN 1230
1300 IF C(1) < > P(1) THEN 1190
1310 IF C(2) < > P(2) THEN 1210
1320 GOTO 1140
1330 IF LEN (I$) < 4 THEN 1380
1340 FOR X = 1 TO 10
1350 IF RIGHT$( I$,LEN (I$) - 2) = P$(X) THEN 1400
1360 IF RIGHT$( I$,LEN (I$) - 3) = P$(X) THEN 1400

```

```

1370 NEXT
1380 PRINT :PRINT "SORRY, I CANNOT GO THERE, SIR. IT
DOES NOT SEEM TO BE IN SHERLOCK'S HOME !"
1390 PRINT :RETURN
1400 IF X = P(1) THEN POP :PRINT :PRINT "SIR, WE SEEM
TO BE IN THE ";P$(X):PRINT "ALREADY !": GOTO
860
1410 IF X < > 6 THEN 1450
1420 GOSUB 1760:IF R > 4 THEN 1450
1430 PRINT :PRINT "SORRY, THE ";P$(6);" IS IN USE AT"
:PRINT "THIS TIME. TRY AGAIN LATER...."
1440 RETURN
1450 P(1) = X
1460 RETURN
1470 IF LEN (I$) < 6 THEN 1520
1480 FOR X = 1 TO 6
1490 IF RIGHT$ (I$,LEN (I$) - 4) = W$(X) THEN 1530
1500 IF RIGHT$ (I$,LEN (I$) - 5) = W$(X) THEN 1530
1510 NEXT
1520 PRINT :PRINT "SORRY, SIR, I DON'T THINK THAT WEA
PON IS IN THIS CASE !":RETURN
1530 IF R(P(1),0) = 0 THEN PRINT :PRINT "SORRY, SIR,
BUT THERE SEEMS TO BE NO WEAPONS IN THIS ROOM
.....PERHAPS SOME- WHERE ELSE ?":RETURN
1540 FOR Y = 1 TO R(P(1),0)
1550 IF R(P(1),Y) = X THEN 1580
1560 NEXT
1570 PRINT :PRINT "SORRY, SIR, BUT I DON'T SEE THAT W
EAPON IN THIS ROOM !":RETURN
1580 T = P(2):P(2) = R(P(1),Y):R(P(1),Y) = T
1590 IF T > 0 THEN RETURN
1600 R(P(1),Y) = R(P(1),R(P(1),0))
1630 R(P(1),0) = R(P(1),0) - 1
1640 RETURN
1650 IF LEN (I$) < 6 THEN 1710
1660 IF P(2) = 0 THEN PRINT :PRINT "SORRY, BUT I AM N
OT CARRYING ANYTHING, SIR !":RETURN
1670 FOR X = 1 TO 6
1680 IF RIGHT$ (I$,LEN (I$) - 4) = W$(X) THEN 1720
1690 IF RIGHT$ (I$,LEN (I$) - 5) = W$(X) THEN 1720
1700 NEXT
1710 PRINT :PRINT "SORRY, SIR, I DON'T THINK THAT WEA
PON IS IN THIS CASE !":RETURN
1720 IF X < > P(2) THEN PRINT :PRINT "SORRY, SIR, BU
T I AM NOT CARRYING THAT WEAPON !":PRINT :RET
URN
1730 R(P(1),0) = R(P(1),0) + 1:R(P(1),R(P(1),0)) = P(
2):P(2) = 0
1740 RETURN
1760 R = INT (RND (1) * 11):IF R < 1 OR R > 10 THEN 1
760
1770 RETURN
1780 VTAB 23:PRINT "PRESS <RETURN> TO CONTINUE...";
1790 GET CH$:IF CH$ < > CHR$(13) THEN 1790
1800 PRINT :HOME :RETURN

```

```
1810 POKE 34,0:CP = PEEK (37):VTAB 1:POKE 35,7:HOME
1820 PRINT "ROOM: ";P$(P(1))
1830 PRINT "CARRYING: ";W$(P(2))
1840 PRINT "VICTIM: ";N$
1850 PRINT :PRINT "VISIBLE: ";
1860 IF R(P(1),0) = 0 THEN 1890
1870 FOR X = 1 TO R(P(1),0)
1880 PRINT W$(R(P(1),X));" ";:NEXT
1890 PRINT :POKE 35,24:POKE 37,CP:POKE 34,7:RETURN
1900 VTAB 20:PRINT "PRESS <RETURN> TO CONTINUE...";
1910 GET CH$:IF CH$ < > CHR$(13) THEN 1910
1920 PRINT :PRINT CHR$(4)"RUN MENU"
```

5. Attack of the Zargons

Due to recently enacted weapons-control laws and your usually lawless attitudes, you are the only person remaining on earth who has a weapon. This puts you in the unique position as the only defender of the planet in the face of an onslaught of invading alien hordes from the planet Zargon. You feel the weight of responsibility and the urgency of the moment as you pull out your patented ACME ALIEN INVADER POWER DRAINER (page 342 in the Whole Planet Catalog) and fearlessly begin firing at the approaching ship.

Such is your power and responsibility in *Attack of the Zargons*. This game uses high-resolution graphics and sounds to create a simple arcade-style game. Shape tables draw the Zargonian ship and your weapon and high-resolution graphics commands produce the other special effects.

A short machine-language sound-generation routine is used to produce the tones heard during the initial "welcome" screen and when shots are fired by the Zargonian ship and your weapon. This routine is the same one used in *Flip Flop*, and it is detailed in that program.

Game Rules

1. If you are selecting the game from the Main Menu, choose 5. If you are not using the menu, run the program.
2. After the game is loaded, there is a short delay before the action begins.
3. Use paddle Ø or the joystick to move your weapon base left and right.
4. Fire at the invader by pressing the game controller button.
5. You receive 100 energy units for hitting the ship.
6. You expend 5 energy units for each shot taken.
7. You lose half of your accumulated energy units if you are hit.
8. You lose half of your accumulated energy units if you do not hit the Zargonian ship at least twice on each screen pass.
9. The game ends when the enemy ship reaches the bottom of the screen.
10. To win, you must have at least 2000 points when the game is over.

Programming Notes

10-30	Program initialization
40	Poke sound routine into memory
50	Welcoming text screen
60-70	Load and initialize shape table
80-110	Game initialization
120-210	Main game loop
140	Draw ship and weapon
150	Zargonian ship fires only 10% of time (maximum)
160	Undraw the Zargonian ship
180	Do score
190	Adjust score if weapon has hit ship less than 2 times across screen.
200	Reset screen-crossing counters
220-230	Game end and check for winning score
240-260	Low score (less than 3000), print message
270-350	High score (3000 or greater), print message
360-390	Another game or end the program and return to the Main Menu
400-420	Score adjustment (add to or subtract from the score)
430-520	Zargon fires at weapon
530-570	Draw weapon at correct position
580-630	Fire at the Zargonian ship
640-670	Musical printing routine
680	Data statements for states

Attack of the Zargons.

```
10 REM ATTACK OF THE ZARGONS
20 M1 = PEEK (115):M2 = PEEK (116)
30 HIMEM: 8192
40 POKE 768,166:POKE 769,1:POKE 770,164:POKE 771,0:
   POKE 772,173:POKE 773,48:POKE 774,192:POKE 775,1
   36:POKE 776,208:POKE 777,253:POKE 778,202:POKE 7
   79,208:POKE 780,245:POKE 781,96
50 HOME :SPEED= 100:A$ = "ATTACK OF THE ZARGONS":HT
   AB 10:GOSUB 640:PRINT :PRINT :A$ = " THEY ARE C
   OMING.....":GOSUB 640:PRINT :PRINT :PRINT
   :A$ = ".....GOOD LUCK !":GOSUB 640:SPEED= 2
   55
60 PRINT "BLOADARC.OBJ"
70 POKE 232,32:POKE 233,78
80 DIM ST$(15):FOR X = 1 TO 15:READ ST$(X):NEXT
90 HOME :T = 0:R1 = INT (PDL (0)):Y = 20:A = 0:B =
   265:C = 5
```

```

100 HGR :ROT= 0:SCALE= 1:XDRAW 2AT R1,150
110 HOME :VTAB 23:HTAB 6:PRINT "ENERGY UNITS STOLEN:
";T
120 FOR Y = 20 TO 130 STEP 5
130 FOR X = A TO B STEP C
140 XDRAW 1AT X,Y:GOSUB 530
150 IF RND (2) < .1 THEN GOSUB 430
160 XDRAW 1AT X,Y
170 NEXT
180 GOSUB 400
190 IF SF < 2 THEN T = INT (T * M):VTAB 23:HTAB 28:C
ALL - 868:PRINT T
200 SF = 0:TP = A:A = B:B = TP:C = - C
210 NEXT
220 TEXT :HOME
230 IF T > 2000 THEN 270
240 PRINT "THIS HAS BEEN A SAD DAY FOR EARTH. ALL":
PRINT "HAS BEEN LOST. THE ZARGONS HAD":PRINT "E
NOUGH POWER LEFT TO DESTROY THE EARTH. "
250 PRINT "YOUR SCORE OF ";T;" JUST WASN'T GOOD"
260 PRINT "ENOUGH.": GOTO 360
270 PRINT "CONGRATULATIONS.....YOU HAVE SUCCESFULLYD
EFENDED THE EARTH AGAINST THE ZARGONS. "
280 PRINT "YOUR SCORE OF ";T;" WAS VERY COMMENDABLE.
":PRINT
290 PRINT "THANKS TO YOUR BRAVERY AND SKILL IN C
OMBAT, YOU HAVE BEEN AWARDED (BY ACT"
300 X1$ = "OF CONGRESS), THE STATE OF "
310 X = INT (RND (1) * 33):IF X < 1 OR X > 15 THEN 3
10
320 X1$ = X1$ + ST$(X):IF LEN (X1$) < 37 THEN X1$ =
X1$ + " IN":IF LEN (X1$) < 35 THEN X1$ = X1$ + "
DEEP":PRINT X1$: GOTO 350
330 PRINT X1$:IF RIGHT$(X1$,2) < > "IN" THEN PRINT
"IN ";
340 IF RIGHT$(X1$,4) < > "DEEP" THEN PRINT "DEEP "
;
PRINT "GRATITUDE."
350 PRINT :PRINT :PRINT "WOULD YOU LIKE TO PLAY AGAI
N (Y/N): ";
370 GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 37
0
380 IF CH$ = "Y" THEN 90
390 POKE 115,M1:POKE 116,M2:PRINT :PRINT CHR$(4)"RU
N MENU"
400 IF T = 0 THEN T = - 2
410 M = .5:IF T < 0 THEN M = 1 + (1 / (ABS (T) / 100
))
420 RETURN
430 HCOLOR= 3:FL = 0
440 Y8 = 155:Y9 = 155:IF X + 1 > R1 - 6 AND X + 1 <
R1 + 6 THEN Y8 = 138:FL = 1
450 IF X + 10 > R1 - 6 AND X + 10 < R1 + 6 THEN Y9 =
138:FL = 1
460 POKE 0,96:POKE 1,10:CALL 768

```

```

470 HPLOT X + 1,Y + 1 TO X + 1,Y8:HPLOT X + 10,Y + 1
    TO X + 10,Y9
480 IF FL = 2 THEN RETURN
490 GOSUB 400
500 IF FL = 1 THEN T = INT ( T * M ):VTAB 23:HTAB 28:C
    ALL - 868:PRINT T
510 HCOLOR= 0:FL = 2
520 GOTO 470
530 R = PDL (0) + 10
540 XDRAW 2AT R1,150
550 R1 = R
560 XDRAW 2AT R1,150
570 IF PEEK ( - 16287) < 128 THEN RETURN
580 U = 0:HCOLOR= 3:T = T - 5
590 IF R1 > = X AND R1 < ( X + 12) THEN U = Y + 1:T
    = T + 100:SF = SF + 1
600 VTAB 23:HTAB 28:CALL - 868:PRINT T
610 HPLOT R1,135 TO R1,U
620 POKE 0,192:POKE 1,10:CALL 768
630 HCOLOR= 0:HPLOT R1,135 TO R1,U:RETURN
640 FOR X = 1 TO LEN (A$)
650 PRINT MID$ (A$,X,1);
660 IF MID$ (A$,X,1) < > " " THEN POKE 0,RND (1) *
    50 + 50:POKE 1,20:CALL 768
670 NEXT :PRINT :RETURN
680 DATA ALASKA,UTAH,NEVADA,ARIZONA,NEW MEXICO,WYOMI
    NG,MONTANA,SOUTH DAKOTA,NORTH DAKOTA,IDAHO,IOWA,
    KANSAS,NEBRASKA,OKLAHOMA,WISCONSIN

```


6. Phaser Practice

Phaser Practice is a game of luck. However, it does show how to choose random points within set boundaries. This is needed to determine where on the target the phaser “blast” is going to hit.

This program also shows how to draw a circle on the screen. The circle is drawn in high-resolution graphics using the `H PLOT TO` statement. Once the first point on the circle’s perimeter is calculated and plotted, the next point is calculated and a line is drawn from the previous point to the new point. This statement, therefore, is similar to a computer version of “connect the dots.”

Game Rules

1. If you are selecting the game from the Main Menu, choose 6. If you are not using the menu, run the program.
2. The directions will be displayed.
3. Enter the number of players. The game allows one to three players.
4. Enter the names of each player as prompted by the program.
5. The target is drawn and each player, in turn, will be prompted for the shot to use.
6. Press a 1, 2, or 3 for your shot.
7. The game ends at the conclusion of a round in which a player’s score is at least 250 points.
8. Upon completion, the players are ranked according to their score. Then you can play again or exit to the Main Menu.

Programming Notes

10	Set high memory to “protect” shape table
20-30	Load and initialize shape table
40-60	Game initialization
70-140	Display instructions
150-170	Get number of players and check for validity
180-210	Get player names and call target-drawing routine
220-230	Game set up

240-470	Main program loop
240-260	Print round number and name (in inverse)
270-300	Get desired shot
310-420	Set parameters for each type of shot
430-450	Show shot and print new score
460-470	Check for winner and loop if none
480-500	Sort names and scores
510-520	Display game results
530-580	Play again or exit to Main Menu
590-760	Plot the individual shot and display it
770-780	Generate a random number between 1 and 10
790-860	Draw, as concentric circles, the target

Phaser Practice.

```

10 M1 = PEEK (115):M2 = PEEK (116):HIMEM: 7499
20 PRINT "BLOADOBJ.PP"
30 POKE 232,76:POKE 233,29
40 S = - 16336
50 T$(1) = "OVERARM":T$(2) = "SIDEARM":T$(3) = "UNDERARM"
60 H$(0) = "MISS !":H$(1) = "10 POINTER !":H$(2) = "20 POINTER !":H$(3) = "30 POINTER !":H$(5) = "BULLSEYE !"
70 HOME :PRINT TAB( 13);"PHASER PRACTICE":PRINT
80 PRINT "THE OBJECT OF THE GAME IS TO BE THE FIRST PLAYER TO ACCUMULATE 250 CREDITS BY FIRING A PHASER AT A TARGET."
90 PRINT :PRINT "THE ZONES ON THE TARGET HAVE POINT VALUES OF 10, 20, 30, AND 50 POINTS."
100 PRINT :PRINT "SHOT TYPE OF SHOT";TAB( 27);"VALUE"
110 PRINT "-----";TAB( 27);"-----"
120 PRINT " 1";TAB( 10);T$(1);TAB( 25);"50 PTS OR MISS"
130 PRINT " 2";TAB( 10);T$(2);TAB( 25);"10, 20 OR 30"
140 PRINT " 3";TAB( 10);T$(3);TAB( 25);"ANYTHING"
150 DIM W(10):R = 0:M = 0
160 PRINT :PRINT "NUMBER OF PLAYERS (1-3): ";
170 GET N$:N = VAL (N$):IF N < 1 OR N > 3 THEN 170
180 PRINT N:PRINT :FOR X = 1 TO N
190 PRINT :PRINT "NAME OF PLAYER #";X;:INPUT " ? ";A$(X)
200 S(X) = 0
210 NEXT :HOME :GOSUB 790
220 VTAB 21:PRINT "ROUND:"
230 POKE 32,14:VTAB 21:FOR X = 1 TO N:PRINT A$(X);:0":NEXT :POKE 32,0

```

```

240 R = R + 1:VTAB 21:HTAB 7:PRINT " ";:HTAB 9:PRI
    NT R
250 FOR X = 1 TO N
260 VTAB (20 + X):HTAB 15:CALL - 868:INVERSE :PRINT
    A$(X);":":NORMAL :PRINT " ";S(X)
270 VTAB 23:PRINT " "":VTAB 23:PRINT "SHO
    T ? ";
280 GET CH$:IF CH$ = CHR$(27) THEN TEXT :HOME : GOT
    O 440
290 T = VAL (CH$):IF T < 1 OR T > 3 THEN 280
300 PRINT T
310 ON T GOTO 320,330,340
320 P1 = .6:P2 = .6:P3 = .6:P4 = .6: GOTO 350
330 P1 = .99:P2 = .77:P3 = .43:P4 = .0001: GOTO 350
340 P1 = .95:P2 = .75:P3 = .45:P4 = .05
350 U = RND (2)
360 IF U > P1 THEN B = 50: GOTO 410
370 IF U > P2 THEN B = 30: GOTO 410
380 IF U > P3 THEN B = 20: GOTO 410
390 IF U > P4 THEN B = 10: GOTO 410
400 B = 0
410 REM
420 S(X) = S(X) + B:VTAB 23:PRINT " "":VT
    AB 23:PRINT H$(B / 10)
430 GOSUB 590
440 VTAB (20 + X):HTAB 15:CALL - 868:PRINT A$(X);":
    ";S(X)
450 NEXT
460 FOR X = 1 TO N
470 IF S(X) < 250 THEN NEXT : GOTO 240
480 FOR X = 1 TO N - 1:FOR Y = X + 1 TO N
490 IF S(X) < S(Y) THEN T = S(X):S(X) = S(Y):S(Y) =
    T:T$ = A$(X):A$(X) = A$(Y):A$(Y) = T$
500 NEXT :NEXT
510 TEXT :HOME :PRINT "WE HAVE A WINNER IN ROUND # "
    ;R
520 PRINT :PRINT "FINAL SCORES:":PRINT :FOR X = 1 TO
    N:PRINT A$(X);" SCORED ";S(X);" POINTS":NEXT
530 PRINT :PRINT :PRINT :PRINT "CARE TO PLAY AGAIN (
    Y/N): ";
540 GET A$:IF A$ < > "Y" AND A$ < > "N" THEN 540
550 IF A$ = "Y" THEN R = 0: GOTO 160
560 HOME :VTAB 8:PRINT "THANX FOR THE GAME !!"
570 POKE 115,M1:POKE 116,M2
580 PRINT :PRINT CHR$(4)"RUN MENU"
590 L = B / 10:IF B = 5 THEN B = 4
600 L = 5 - L
610 I = INT (RND (2) * 361):IF I < 1 OR I > 359 THEN
    610
620 Y1 = L * SIN (I / 57.3) * 20:X1 = L * COS (I / 5
    7.3) * 20
630 L = L - 1:Y2 = L * SIN (I / 57.3) * 20:X2 = L *
    COS (I / 57.3) * 20
640 Y1 = INT (80 - Y1):Y2 = INT (80 - Y2):X1 = INT (
    140 + X1):X2 = INT (140 + X2)

```

```

650 GOSUB 770:Y3 = Y1 - RN:IF Y3 < Y2 THEN Y3 = Y2
660 GOSUB 770:X3 = X1 - RN:IF X3 < X2 THEN X3 = X2
670 HCOLOR= 3:HPOINT 0,160 TO X3,Y3:HPOINT 279,160 TO
X3,Y3
680 SCALE= 2:ROT= 0:FOR Z = 1 TO 10
690 XDRAW 1AT X3,Y3
700 FOR Z1 = 1 TO 6
710 SO = PEEK (S):SO = PEEK (S):SO = PEEK (S)
720 NEXT
730 XDRAW 1AT X3,Y3
740 NEXT
750 HCOLOR= 0:HPOINT 0,160 TO X3,Y3:HPOINT 279,160 TO
X3,Y3
760 RETURN
770 RN = INT (RND (2) * 11):IF RN > 10 OR RN < 1 THE
N 770
780 RETURN
790 HGR
800 HCOLOR= 3
810 FOR Z = 20 TO 80 STEP 20
820 FOR I = 0 TO 360 STEP 4
830 Y = INT (80 - Z * SIN (I / 57.3)):X = INT (140 +
Z * COS (I / 57.3))
840 IF I = 0 THEN HPOINT X,Y
850 HPOINT TO X,Y
860 NEXT :NEXT :RETURN

```

7. Acey-Ducey

This is a game for gamblers. Three cards are picked at random and you bet on whether or not the value of the third card will be between the first two.

The cards are displayed on the screen using subroutines to draw the blank cards and the card's symbols, correctly positioned. The card symbols—hearts, diamonds, spade, and clubs—are drawn using a shape table.

One suggestion for improving the game is to add a real “deck” and deal from it. As the game is presently programmed, the cards are drawn at random from an “unlimited” deck; a card, for example, the ten of hearts, can occur more than once in a game, or even in one hand.

Game Rules

1. If you are selecting the game from the Main Menu, choose 7. If you are not using the menu, run the program.
2. The instructions will be displayed.
3. Press <RETURN> to start the game.
4. You begin with a \$100.00 stake.
5. Two cards will be shown face up and one face down.
6. Enter your bet. You can bet nothing (by entering 0) or up to your full stake.
7. The face-down card will be shown.
8. If the third card's value is between the values of the first two cards, you win the amount of your wager. If the third card's value is less than the first card, greater than the second card, or equal to either of the first two cards, you lose the amount of your wager.
9. Card ranking is 2 through 10, Jack, Queen, King, Ace.
10. The game ends when you have no more money.
11. When the game ends, you can play again or exit to the Main Menu.

Programming Notes

1Ø	Set high-memory boundary
2Ø	Initialize and load shape table
3Ø-6Ø	Program initialization
7Ø-12Ø	Dollars and cents display routine
13Ø-14Ø	Rounding routine
15Ø-195	Display directions
20Ø-21Ø	Wait for a <RETURN> before proceeding
22Ø-27Ø	Initialize screen and display
28Ø	Check for player with no money
29Ø-31Ø	Pick first two cards
32Ø	Redo blank cards on screen
33Ø-35Ø	Pick suit for first card
36Ø	Show first card
37Ø-39Ø	Pick suit for second card
40Ø	Show second card
41Ø-46Ø	Get wager; exit or check for validity of wager
47Ø-50ØØ	Pick third card and suit
51Ø	Show third card
52Ø	Does the bet win?
53Ø-55Ø	Yes, so add to your stake
56Ø-57Ø	No, so decrease your stake
58Ø	Short pause and loop for next hand
59Ø-60ØØ	Out of money, therefore game is over
61Ø-66Ø	Play again or exit to Main Menu
67Ø-74Ø	Print blank cards on screen (no markings)
120Ø-123Ø	Display markings for two
130Ø-136Ø	Display markings for three
140Ø-148Ø	Display markings for four
150Ø-157Ø	Display markings for five
160Ø-165Ø	Display markings for six
170Ø-178Ø	Display markings for seven
180Ø-186Ø	Display markings for eight
190Ø-197Ø	Display markings for nine
200Ø-204Ø	Display markings for ten
210Ø-212Ø	Display markings for Jack
220Ø-222Ø	Display markings for Queen
230Ø-232Ø	Display markings for King
240Ø-242Ø	Display markings for Ace

Acey-Ducey.

```

10 M1 = PEEK (115):M2 = PEEK (116):HIMEM: 8192
20 DIM C$(14):POKE 232,32:POKE 233,78:PRINT "BLOOD
CARDS"
30 DATA 2,3,4,5,6,7,8,9,10,JACK,QUEEN,KING,ACE
40 FOR X = 2 TO 14:READ C$(X):NEXT
50 ROT= 0:SCALE= 1
60 GOTO 150
70 Q = INT (Q * 100 + .5) / 100
80 Q$ = STR$(Q):IF Q = 0 THEN Q$ = ""
90 IF INT (Q) = 0 THEN Q$ = "0" + Q$
100 IF Q = INT (Q) THEN Q$ = Q$ + ".00": GOTO 120
110 IF ASC (RIGHT$(Q$,3)) < > 46 THEN Q$ = Q$ + "0"
"

120 RETURN
130 R = INT (RND (1) * 17):IF R < 2 OR R > 14 THEN 1
30
140 RETURN
150 HOME
160 PRINT "ACEY-DUCEY IS PLAYED IN THE FOLLOWING":PR
INT "WAY:":PRINT :PRINT "THE DEALER (COMPUTER) D
EALS TWO CARDS"
170 PRINT "FACE UP. YOU HAVE THE OPTION TO BET OR N
OT TO BET DEPENDING ON WHETHER OR NOT YOU THINK
THE NEXT CARD DEALT WILL HAVE A VALUE BETWEEN T
HE FIRST TWO."
180 PRINT :PRINT "IF YOU DO NOT WISH TO PLACE A BET,
":PRINT "ENTER '0' FOR YOUR BET.":PRINT
190 PRINT "TIES WITH YOUR UPPER AND LOWER CARDS AREA
UTOMATICALLY WON BY THE HOUSE."
195 PRINT :PRINT "ENTERING 'QUIT' AS YOUR BET WILL A
LLOW YOU TO END THE GAME EARLY."
200 VTAB 23:PRINT "PRESS <RETURN> TO CONTINUE...";
210 GET I$:IF I$ < > CHR$(13) THEN 210
220 HOME :HGR
230 VTAB 21:PRINT "STAKE:"
240 VTAB 22:PRINT "YOUR WAGER: $"
250 Q = 100
260 GOSUB 70
270 VTAB 21:HTAB 8:CALL - 868:PRINT Q$
280 IF Q < .02 THEN 590
290 GOSUB 130:A = R
300 GOSUB 130:B = R
310 IF A > = B - 1 THEN 290
320 GOSUB 670
330 X = 20
340 GOSUB 130:IF R > 4 THEN 340
350 ST = R
360 ON (A - 1)GOSUB 1200,1300,1400,1500,1600,1700,18
00,1900,2000,2100,2200,2300,2400
370 X = 100
380 GOSUB 130:IF R > 4 THEN 380
390 ST = R
400 ON (B - 1)GOSUB 1200,1300,1400,1500,1600,1700,18
00,1900,2000,2100,2200,2300,2400

```

```

410 VTAB 22:HTAB 14:CALL - 958:INPUT " ";M$
415 IF M$ = "QUIT" THEN TEXT :HOME :PRINT "THANKS FO
R PLAYING!":PRINT : GOTO 610
420 M = VAL (M$):IF M < 0 THEN PRINT CHR$ (7);: GOTO
410
430 IF M = 0 THEN VTAB 23:CALL - 868:PRINT "CHICKEN
!": GOTO 270
440 IF M$ = Q$ THEN 470
450 IF M < = Q THEN 470
460 VTAB 23:CALL - 868:PRINT "SORRY, BUT YOU ONLY H
AVE $";Q$: GOTO 280
470 GOSUB 130:C = R
480 GOSUB 130:IF R > 4 THEN 480
490 ST = R
500 X = 180
510 ON (C - 1)GOSUB 1200,1300,1400,1500,1600,1700,18
00,1900,2000,2100,2200,2300,2400
520 IF C < = A OR C > = B THEN 560
530 VTAB 23:CALL - 868:PRINT "YOU WIN !!!!!"
540 Q = Q + M
550 GOTO 580
560 VTAB 23:CALL - 868:PRINT "SORRY, BUT YOU LOSE !
"
570 Q = Q - M
580 FOR DE = 1 TO 2500:NEXT : GOTO 260
590 TEXT :HOME
600 PRINT "SORRY, FRIEND, BUT YOU LOST YOUR ENTIRE S
TAKE!":PRINT
610 PRINT "TRY AGAIN (Y/N): ";
620 GET A$:IF A$ < > "Y" AND A$ < > "N" THEN 620
630 IF A$ = "Y" THEN 220
640 PRINT "HOPE YOU HAD FUN!!!"
650 POKE 115,M1:POKE 116,M2
660 PRINT :PRINT CHR$ (4)"RUN MENU"
670 HGR
680 HCOLOR= 3
690 FOR X = 20 TO 80
700 HPLOT X,10 TO X,75
710 HPLOT X + 80,10 TO X + 80,75
720 HPLOT X + 160,10 TO X + 160,75
730 NEXT
740 RETURN
1200 X = X + 30
1210 Y = 38:XDRAW STAT X,Y
1220 Y = 56:XDRAW STAT X,Y
1230 RETURN
1300 X = X + 15
1310 Y = 34:XDRAW STAT X,Y
1320 X = X + 15
1330 Y = 47:XDRAW STAT X,Y
1340 X = X + 15
1350 Y = 62:XDRAW STAT X,Y
1360 RETURN
1400 X = X + 20
1410 Y = 38:XDRAW STAT X,Y

```



```

1420 X = X + 20
1430 XDRAW STAT X,Y
1440 X = X - 20
1450 Y = 56:XDRAW STAT X,Y
1460 X = X + 20
1470 XDRAW STAT X,Y
1480 RETURN
1500 FOR Z = X + 15 TO X + 45 STEP 15
1510 IF Z = X + 30 THEN 1550
1520 Y = 34:XDRAW STAT Z,Y
1530 Y = 62:XDRAW STAT Z,Y
1540 GOTO 1560
1550 Y = 47:XDRAW STAT Z,Y
1560 NEXT
1570 RETURN
1600 FOR Z = X + 20 TO X + 40 STEP 20
1610 Y = 34:XDRAW STAT Z,Y
1620 Y = 47:XDRAW STAT Z,Y
1630 Y = 62:XDRAW STAT Z,Y
1640 NEXT
1650 RETURN
1700 FOR Z = X + 15 TO X + 45 STEP 15
1710 IF Z = X + 30 THEN 1760
1720 Y = 34:XDRAW STAT Z,Y
1730 Y = 47:XDRAW STAT Z,Y
1740 Y = 62:XDRAW STAT Z,Y
1750 GOTO 1770
1760 Y = 47:XDRAW STAT Z,Y
1770 NEXT
1780 RETURN
1800 FOR Z = X + 20 TO X + 40 STEP 20
1810 Y = 31:XDRAW STAT Z,Y
1820 Y = 42:XDRAW STAT Z,Y
1830 Y = 53:XDRAW STAT Z,Y
1840 Y = 64:XDRAW STAT Z,Y
1850 NEXT
1860 RETURN
1900 FOR Z = X + 15 TO X + 45 STEP 15
1910 IF Z = X + 30 THEN Y = 47:XDRAW STAT Z,Y: GOTO 1
960
1920 Y = 31:XDRAW STAT Z,Y
1930 Y = 42:XDRAW STAT Z,Y
1940 Y = 53:XDRAW STAT Z,Y
1950 Y = 64:XDRAW STAT Z,Y
1960 NEXT
1970 RETURN
2000 FOR Y = 29 TO 65 STEP 9
2010 X1 = X + 20:XDRAW STAT X1,Y
2020 X1 = X + 40:XDRAW STAT X1,Y
2030 NEXT
2040 RETURN
2100 X = X + 25:Y = 42
2110 XDRAW 5AT X,Y
2120 RETURN
2200 X = X + 25:Y = 42

```

```
2210 XDRAW 6AT X,Y
2220 RETURN
2300 X = X + 25:Y = 47
2310 XDRAW 7AT X,Y
2320 RETURN
2400 X = X + 30:Y = 47
2410 XDRAW STAT X,Y
2420 RETURN
```

8. Big Government

Ten score and eight years ago, our forefathers had a small government. Now you have a BIIIIIIIG (that's *really* big!) government. You are the president of a country and must decide how to run the party and keep your constituents (the unsuspecting voters who elected you) happy. You will be in office for 48 months (unless you are impeached before then).

You can buy and sell bonds for the treasury, spend money (what some politicians seem to do best) to keep the people happy, and solicit contributions (what some politicians do second best). But watch out! Graft, kickbacks, and unforeseen events always occur when the future looks brightest.

You will be constantly updated on the latest events by the news teletype (notice the percussive sound of the hammer hitting the keys) and your ever vigilant advisers, who interrupt periodically with news flashes.

If you have hopes and dreams for high office, choose 8 from the Main Menu and play *Big Government*. But beware, the public is fickle at best and downright hostile at worst!

Game Rules

1. If you are selecting the game from the Main Menu, choose 8. If you are not using the menu, run the program.
2. The instructions will be displayed.
3. Enter your name when prompted. You can enter your entire name, but the screen format is "neater" if you use only your first name.
4. Each month until the election will bring new plans, conditions, and perils.
5. You can buy and sell bonds at the current market price. This varies from month to month.
6. You should spend approximately \$20.00 per voter to keep them in your political camp. Spending more than that results in more people joining your forces, while spending less alienates a number of voters.
7. You can alienate a small number of voters, but alienating too many at one time will result in your impeachment and subsequent removal from office.

8. The amount you spend on soliciting contributions can be up to twice the number of bonds you have in the treasury.
9. You need at least one person in your camp for every \$10.00 spent on solicitation. After all, you don't want to overwork your supporters.
10. Soliciting contributions is the primary way to make money.
11. The game ends at election time. This is 48 months, or turns, from the beginning of the game (unless you are impeached).
12. Upon completing the game, the statistics for your term in office will be reviewed and the game will end.

Programming Notes

10-50	Instructions
60-70	Get player's name
80-130	Game initialization and screen setup
140-210	Begin printing monthly status
220-250	Crisis — half of the voters leave camp
260-300	Continue printing monthly status
310	Check for end of game and branch if finished
320	Determine price for bonds
330	Print bond price
340-380	Allow player to buy bonds and adjust figures
390-430	Allow player to sell bonds and adjust figures
440-480	Player specifies amount to spend on voters
490-560	Player specifies amount for solicitation
570-700	Adjust monthly figures by random amounts
710-750	Crisis — alienated too many people
760	Deficit-spending error message
770-810	Screen status refresh routine
820	Generate random number between 1 and 8
830-850	Crisis — mental incompetence
860-910	End-of-game status report
920-980	Branching logic for game performance
990	Excellent job message
1000-1010	Poor job message
1020-1040	Fair job message
1050-1070	End the game
1080-1090	Bad job message
1100-1120	Good money raiser message

113Ø-115Ø Pick random News Flash message
 116Ø-119Ø Print News Flash message
 120Ø-122Ø News Flash — dishonest brother
 123Ø-125Ø News Flash — cabinet resigns
 126Ø-128Ø News Flash — economic policies
 129Ø-132Ø News Flash — campaign practices
 133Ø-135Ø News Flash — Mideast breakthrough
 136Ø-138Ø News Flash — inflation down
 139Ø-141Ø News Flash — oil prices up
 142Ø-144Ø News Flash — polls down
 145Ø-147Ø News Flash — campaign contribution
 148Ø-151Ø Teletype printing routine

Big Government.

```

1Ø REM          BIG GOVERNMENT
2Ø HOME :PRINT TAB( 13);"BIG GOVERNMENT":PRINT :PRI
   NT :PRINT
3Ø PRINT " YOU ARE THE PRESIDENT (WE WON'T SAY OFW
   HICH NEARBY COUNTRY) AND YOU HAVE TO MAKE THE
   EVERYDAY DECISIONS THAT WILL"
4Ø PRINT "ENSURE THAT YOU WILL BE RE-ELECTED NEXT E
   LECTION (WHICH HAPPENS TO BE 48 MONTHS FROM NOW.
   )":PRINT :PRINT
5Ø PRINT " YOU CAN MAKE MONEY FOR THE PARTY (
   WHICH IS ONE OF YOUR MANY JOBS) BY BUYING AN
   D SELLING BANK BONDS, BUT BE CAREFUL NOT TO AL
   IENATE TOO MANY VOTERS,BECAUSE THEY DO ALL OF YO
   UR WORK AND ARETHE KEY TO YOUR SUCCESS."
6Ø VTAB 23:INPUT "PLEASE ENTER YOUR NAME: ";N$
7Ø IF LEN (N$) = Ø THEN VTAB 23:CALL - 868: GOTO 6
   Ø
8Ø HOME :VTAB 1:PRINT "TREASURY FUNDS:":PRINT "BOND
   S IN BANK:":PRINT "FAVORABLE VOTERS:":PRINT "MON
   THS TIL ELECTION:"
9Ø VTAB 5:PRINT "<---> <---> <---> <---> <---> <--->
   <--->"
1ØØ VTAB 2Ø:POKE 34,6:HOME
11Ø D1 = Ø:P1 = Ø:Z = 48:P = 100 - INT (RND (1) * 2Ø
   ):H = INT (RND (1) * 3000):E = INT (RND (1) * 39
   1):S = 3000 + H - E:Y = 3:A = INT (H / Y):I = 5:
   Q = 1
12Ø BR = INT (A / P)
13Ø GOSUB 77Ø: GOTO 22Ø
14Ø PRINT :S$ = "MR. PRESIDENT, OUR OFFICIAL SOURCES
   ":GOSUB 148Ø
15Ø P = P + I:Z = Z - 1:GOSUB 77Ø
16Ø S$ = "REPORT THAT ONLY " + STR$ (Z) + " MONTHS R
   EMAIN":GOSUB 148Ø
17Ø S$ = "UNTIL ELECTION TIME, AND " + STR$ (D) + "
   VOTERS":GOSUB 148Ø
  
```

```

180 S$ = "WERE ALIENATED. HOWEVER, " + STR$(I) + "
    VOTERS":GOSUB 1480
190 S$ = "HAVE JOINED OUR SIDE.":GOSUB 1480
200 IF RND (1) < .2 THEN GOSUB 1130:GOSUB 770
210 IF FL THEN PRINT :S$ = "AN ANONYMOUS CONTRIBUTOR
    HAS JUST DONAT-ED $" + STR$(FL):GOSUB 1480:FL
    = 0
220 IF Q > 0 THEN 260
230 FL = 1
240 P = INT (1 + P / 2):GOSUB 770
250 PRINT :S$ = "BAD NEWS, MR. PRESIDENT. A RECENT
    NEWS POLL SHOWS THAT YOU HAVE LOST HALF OF YOU
    R POTENTIAL VOTERS DUE TO A SURPRISE ANNOUNCEMEN
    T FROM YOUR OPPOSITION.":GOSUB 1480
260 PRINT :S$ = "THE NUMBER OF PEOPLE IN YOUR CAMP N
    OW STANDS AT " + STR$(P) + ".":GOSUB 1480
270 PRINT :S$ = "THE PARTY NOW HAS " + STR$(A) + "
    BANK BONDS.":GOSUB 1480
280 PRINT :S$ = "YOU RECEIVED $" + STR$(H) + " IN D
    ONATIONS.":GOSUB 1480
290 IF E < > 0 THEN PRINT :S$ = "KICKBACKS PAID OUT
    AMOUNTED TO $" + STR$(E):GOSUB 1480
300 PRINT :S$ = "YOU NOW HAVE $" + STR$(S) + " IN T
    HE TREASURY":GOSUB 1480:PRINT
310 IF Z = 0 THEN 860
320 C = INT (RND (1) * 11):Y = C + 17
330 S$ = "BONDS ARE NOW SELLING AT $" + STR$(Y) + "
    .":GOSUB 1480:PRINT :PRINT
340 S$ = "HOW MANY BONDS DO YOU WISH TO BUY? ":FQ =
    1:GOSUB 1480
350 INPUT "":Q$:Q = VAL (Q$):IF Q < 0 THEN 830
360 IF Y * Q > S THEN GOSUB 760:GOTO 340
370 A = A + Q:S = S - Y * Q:C = 0
380 PRINT :IF Q > 0 THEN GOSUB 770
390 S$ = "HOW MANY BONDS DO YOU WISH TO SELL? ":FQ =
    1:GOSUB 1480
400 INPUT "":Q$:Q = VAL (Q$):IF Q < 0 THEN 830
410 IF A < Q THEN PRINT :S$ = "MR. PRESIDENT, THE PA
    RTY ONLY HAS " + STR$(A):GOSUB 1480:S$ = "BONDS
    IN THE BANK.":GOSUB 1480:PRINT :GOTO 390
420 A = A - Q:S = S + Y * Q:C = 0
430 PRINT :IF Q > 0 THEN GOSUB 770
440 S$ = "HOW MUCH MONEY SHOULD BE SPENT TO KEEP OU
    R VOTERS HAPPY? ($20 PER VOTER USUALLYDOES IT):
    ":FQ = 1:GOSUB 1480
450 INPUT "":Q$:Q = VAL (Q$):IF Q < 0 THEN 830
460 IF Q > S THEN GOSUB 760:GOTO 440
470 S = S - Q:C = 1:PRINT
480 GOSUB 770
490 S$ = "HOW MUCH DO YOU WISH TO SPEND ON":GOSUB 14
    80:S$ = "SOLICITING DONATIONS? ":FQ = 1:GOSUB 14
    80
500 INPUT "":D$:D = VAL (D$):IF D = 0 THEN 570
510 IF D < 0 THEN 830
520 IF D > (2 * A) THEN PRINT :S$ = "MR. PRESIDENT,

```

```

YOU KNOW WE NEED MORE      BONDS IN THE BANK TO BA
CK THAT TYPE OF  SOLICITATION VENTURE.  PLEASE R
ECONSIDERTHE AMOUNT.":GOSUB 1480:PRINT : GOTO 49
0
530  IF D > S THEN GOSUB 760: GOTO 490
540  IF D < = 10 * P THEN 560
550  PRINT :S$ = "BUT YOU ONLY HAVE " + STR$ (P) + "
    PEOPLE TO DO":GOSUB 1480:S$ = "THE SOLICITING.
    EACH PERSON CAN ONLY SPEND $10 FOR THE PROJECT."
    :GOSUB 1480: GOTO 490
560  S = S - D:GOSUB 770
570  GOSUB 820
580  Y = C:H = D * Y:E = 0:IF C > 6 THEN FL = C * P:H
    = H + FL
590  GOSUB 820
600  IF C < 4 THEN E = INT (S / C)
610  S = S - E + H:GOSUB 770
620  GOSUB 820
630  IF P = 0 THEN P = 1
640  I = INT ((RND (1) * P) / 2) + 1
650  C = INT (Q / 20)
660  Q = INT (RND (1) * 101):Q = INT (Q / 15)
670  D = P - C:IF D < = 0 THEN I = I + INT (ABS (D)
    / 19 * RND (1) * 11):D = 0: GOTO 140
680  IF D * 20 > 9 * P THEN 710
690  P1 = INT (D1 * 110 / (P * Z))
700  P = C:D1 = D1 + D: GOTO 140
710  PRINT :PRINT "YOU ALIENATED ";D;" PEOPLE IN ONE"
    :PRINT "MONTH 1":PRINT
720  PRINT "DUE TO EXTREME MISMANAGEMENT YOU HAVE ":P
    RINT "NOT ONLY BEEN IMPEACHED "
730  PRINT "AND THROWN OUT OF OFFICE BUT YOU HAVE"
740  PRINT "ALSO BEEN DECLARED NATIONAL FINK."
750  GOTO 1050
760  S$ = "MR. PRESIDENT, WE ONLY HAVE $" + STR$ (S):
    GOSUB 1480:S$ = "IN THE TREASURY. WE CANNOT COND
    ONE":GOSUB 1480:S$ = "DEFICIT SPENDING.":GOSUB 1
    480:PRINT :RETURN
770  VTAB 1:HTAB 18:CALL - 868:PRINT S
780  VTAB 2:HTAB 16:CALL - 868:PRINT A
790  VTAB 3:HTAB 19:CALL - 868:PRINT P
800  VTAB 4:HTAB 22:CALL - 868:PRINT Z
810  VTAB 24:RETURN
820  C = INT (RND (1) * 8) + 1:RETURN
830  PRINT :PRINT "SORRY, MR. PRESIDENT, BUT I CANNOT
    DO WHAT YOU WISH.":PRINT
840  PRINT "YOU HAVE BEEN IMPEACHED ON GROUNDS OF M
    ENTAL INCOMPETENCE. YOUR TERM IS HERE-BY ENDED.
    ":PRINT
    GOTO 1050
850  HOME
860  S$ = "IN YOUR 48-MONTH TERM OF OFFICE, YOU HA
    VE OFFENDED OVER " + STR$ (D1) + " PEOPLE.":GOSU
    B 1480:PRINT
880  IF P = 0 THEN P = 1

```

```

890 L = INT ( A / P )
900 S$ = "YOU BEGAN WITH A RATIO OF " + STR$ ( BR ) +
" BANK":GOSUB 1480:S$ = "BONDS TO EVERY VOTER, A
ND YOU ENDED WITH A RATIO OF " + STR$ ( L ) + " BON
DS PER VOTER.":GOSUB 1480
910 PRINT
920 IF D1 > P THEN PRINT : GOTO 720
930 IF P > 110 AND A > 1100 AND S > 2500 THEN 1100
940 IF L < 7 THEN 1080
950 IF P1 > 10 THEN 1000
960 IF L < 10 AND D1 > 30 THEN 1000
970 IF P1 > 3 THEN 1020
980 IF L < 10 THEN 1020
990 PRINT "A TRULY REMARKABLE PERFORMANCE !!":PRINT
"SURELY NONE OF YOUR PREDECESSORS COULD HAVE DO
NE A MORE NOBLE JOB.": GOTO 1050
1000 PRINT "YOUR UNFEELING, AUTOCRATIC RULE IS R
EMINISCENT OF CHICAGO DURING THE AGE OF BOSS TWEE
D."
1010 PRINT "THE VOTERS (THAT ARE STILL LEFT) FIND Y
OU A VERY UNLIKABLE GUY. YOU WILL BE LUCKY TO
GET ANY JOB IN THIS COUNTRY IN THE FUTURE.": GOT
O 1050
1020 PRINT "WELL, YOU DIDN'T MAKE THE HALL OF FAME, B
UT YOU DIDN'T MAKE MANY ENEMIES EITHER."
1030 PRINT INT ((P * 8) * (1 + RND (1) * 13) / 1000);
" PEOPLE WOULD LIKE TO SEE YOU DEAD,"
1040 PRINT "BUT WE ALL HAVE LITTLE PROBLEMS NOW AND T
HEN."
1050 PRINT "":PRINT "SO LONG FOR NOW."
1060 FOR J = 1 TO 10000:NEXT
1070 PRINT "RUN MENU"
1080 PRINT "POOR JOB, ";N$;".":PRINT "YOU WERE SUPPOS
ED TO WORK AT THE JOB, NOT JUST SIT BACK AND T
AKE THE PAY."
1090 GOTO 1050
1100 PRINT "YOUR PARTY'S BANKBOOK GREW QUITE A BIT. Y
OU'RE AN EXCELLENT PRESIDENT, BUT YOUR DETRAC TOR
S ARE ASKING FOR A"
1110 PRINT "CONGRESSIONAL REVIEW OF YOUR FINANCES. G
OOD LUCK."
1120 GOTO 1050
1130 REM CAMPAIGN DISASTERS
1140 CD = INT ( RND (1) * 10 ):IF CD < 1 OR CD > 9 THEN
1140
1150 ON CDGOSUB 1200,1230,1260,1290,1330,1360,1390,14
20,1450
1160 S$ = "***** NEWS FLASH *****"
+ CD$
1170 PRINT "":PRINT :PRINT :GOSUB 1480
1180 PRINT :PRINT
1190 RETURN
1200 CD$ = " YOUR BROTHER WAS FOUND TO BE IN C
AHOOTS WITH A SPY RING FROM BOTSWANA. YOU HAVE
LOST ONE HALF OF YOUR VOTERS AND ONE THIRD OF
YOUR BONDS."

```



```

1210 A = INT (A * .66):P = INT (P / 2):D = D + P
1220 RETURN
1230 CD$ = " TODAY THE ENTIRE PRESIDENTIAL CABINET R
ESIGNED IN A MASSIVE DISAGREEMENT OVER DOMESTIC
POLICY. YOU LOSE HALF YOUR VOTERS."
1240 P = INT (P / 2):D = D + P
1250 RETURN
1260 CD$ = " THE NEW ECONOMIC POLICIES ANNOUNCED T
ODAY BY THE PRESIDENT HAVE PRECIPITATED A FALLING
-OUT BY BANKERS, AND THE PRESIDENT HAS LOST HAL
F OF HIS BANK BONDS AS A RESULT OF THE REACTION.
"
1270 A = INT (A / 2)
1280 RETURN
1290 CD$ = " THE CAMPAIGN PRACTICES OF THE PRESI- D
ENT CAME UNDER CLOSE SCRUTINY TODAY BY THE NATIO
NAL BOARD OF REVIEW (N.B.R) AND THE PRESIDENT
'S PARTY HAS BEEN FINED $"
1300 CD$ = CD$ + STR$ (INT (S / 3)) + "."
1310 S = S - INT (S / 3)
1320 RETURN
1330 CD$ = " THE PRESIDENT HAS SUCCESSFULLY PRE- V
ENTED A MID-EAST CRISIS BY HIS SKILLFUL HANDLING
OF FOREIGN AFFAIRS. THIS IS EXPECTED TO HAVE
A GOOD EFFECT ON HIS RE-ELECTION DRIVE."
1340 P = INT (P * 1.4)
1350 RETURN
1360 CD$ = " RECENTLY RELEASED REPORTS SHOW THAT T
HE RATE OF INFLATION HAS SLOWED CONSIDERABLY DU
E TO THIS ADMINISTRATION'S EFFORTS TO REDUCE
GOVERNMENT SPENDING."
1370 P = INT (P * 1.2)
1380 RETURN
1390 CD$ = " THE ARABS HAVE AGAIN RAISED OIL PRICES I
N RETALIATION TO GOVERNMENT CONCESSIONS IN THE MI
D-EAST. THIS IS EXPECTED TO REFLECT IN PRESID
ENTIAL SUPPORT AT THE POLLS THIS FALL."
1400 P = INT (P * .8)
1410 RETURN
1420 CD$ = " RECENT PUBLIC OPINION POLLS SHOW THE P
RESIDENT HAS HAD A 10% DROP IN VOTER SUPPORT."
1430 P = INT (P * .9)
1440 RETURN
1450 CD$ = " TODAY, CHARLES T. FARNSWORTH, THE N
OTED TEXAS MILLIONAIRE, HAS ANNOUNCED HIS SUPPO
RT FOR THE PRESIDENT AND CON- TRIBUTED $4500 TO
THE PARTY AT A MORNING PRESS CONFERENCE."
S = S + 4500:P = P + 1
1460 RETURN
1470
1480 SPEED= 137:FOR X = 1 TO LEN (S$):PRINT MID$ (S$,
X,1):IF MID$ (S$,X,1) = " " THEN 1500
1490 SOUND = PEEK ( - 16336)
1500 NEXT :IF NOT FQ THEN PRINT
1510 FQ = 0:SPEED= 255:RETURN

```

[The text on this page is extremely faint and illegible. It appears to be a list or a series of entries, possibly containing names and dates, but the characters are too light to transcribe accurately.]

9. Tic-Tac-Toe

Remember sitting for hours when you were a child, playing Tic-Tac-Toe with whoever would take the time to sit and play with you? Well, now you have a partner who is always ready, willing, and quite able (to beat you). This is as good an excuse as any to start your second or third childhood (many of us go through it, don't we?).

With the computer as your opponent you may have met your match, because you will find that the computer plays quite well. The computer even doubles as scorekeeper, keeping track of who wins which games, including those that go to the proverbial "cat."

This game uses only text, but it can be easily modified to use graphics by changing the display routine in line 69Ø. This is the only line that would have to be changed.

This game simulates how a person thinks. The computer evaluates the current board conditions and makes a logical choice as to which move to make. This is one reason why children enjoy playing this simple game—it encourages them to think logically. The choices are based on a mathematical analysis of the possible winning positions. This analysis is performed in lines 19Ø through 51Ø.

Game Rules

1. If you are selecting the game from the Main Menu, choose 9. If you are not using the menu, run the program.
2. Choose X's or O's by pressing the X or O key.
3. Specify if you want to move first by answering Yes (Y) or No (N) to the prompt.
4. When it is your turn, signify the position you want to move to by pressing the number associated with that position. Positions are numbered a follows:

1	2	3
4	5	6
7	8	9
5. The object of the game is to get three of your markers in a row — horizontally, vertically, or diagonally.

6. If neither you nor the computer gets three markers in a row, the "cat" wins the game.
7. At the game's completion, you can play again or exit to the Main Menu.

Programming Notes

10-30	Initialization and screen preparation
40-50	Prompt for you to choose a marker
60	Assignment of X and O to you and the computer
70-80	Prompt for you to choose who goes first
90	Screen-instruction setup
100	Screen display if you move first
110-120	Selection of position if computer moves first
130-160	Get your move and see if it's valid
170	Check for a winner
180	Set arrays to determine the winner
190-510	Select computer's move
520	Set computer move, display results, check for winner, and loop back for next move
600-680	Determine array values for judging the winner
690	Display playing board
700-810	Determine if there is a winner and display appropriate message
820-830	Prompt to play again or exit to Main Menu
840-850	End current game

Tic-Tac-Toe.

```

10  REM      TIC-TAC-TOE
20  FOR X = 1 TO 9:B(X) = 0:NEXT :M$(0) = "."
30  HOME :PRINT TAB( 15);"TIC-TAC-TOE"
40  PRINT :PRINT "WOULD YOU LIKE X'S OR O'S: ";
50  GET M$:IF M$ < > "X" AND M$ < > "O" THEN 50
60  PRINT M$:M$(9) = M$:M$(1) = "X":IF M$ = "X" THEN
    M$(1) = "O"
70  PRINT :PRINT "WOULD YOU LIKE TO MOVE FIRST (Y/N)
    : ";
80  GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 80
90  PRINT CH$:HOME :PRINT "POSITIONS ARE NUMBERED 1
    THROUGH 9 FROM THE TOP-LEFT TO BOTTOM-RIGHT CORN
    ER.":VTAB 15:POKE 34,5:PRINT :HOME
100 IF CH$ = "Y" THEN GOSUB 690: GOTO 130
110 X = INT (RND (2) * 10):IF X < 1 OR X > 9 THEN 11
    0
120 B(X) = 1:GOSUB 690
130 VTAB 20:PRINT "YOUR MOVE (1-9): ";

```

```

140 GET X$:X = VAL (X$):IF X < 1 OR X > 9 THEN 140
150 PRINT X:IF B(X) < > 0 THEN PRINT CHR$(7):GOTO
    130
160 B(X) = 9
170 GOSUB 700:REM SEE IF WINNER
180 GOSUB 600
190 REM
200 REM SELECT A MOVE
210 REM
220 CF = 2
230 IF R(1) < > CF THEN 260
240 FOR X = 1 TO 3:IF B(X) = 0 THEN 520
250 NEXT
260 IF R(2) < > CF THEN 290
270 FOR X = 4 TO 6:IF B(X) = 0 THEN 520
280 NEXT
290 IF R(3) < > CF THEN 320
300 FOR X = 7 TO 9:IF B(X) = 0 THEN 520
310 NEXT
320 IF R(4) < > CF THEN 350
330 FOR X = 1 TO 7 STEP 3:IF B(X) = 0 THEN 520
340 NEXT
350 IF R(5) < > CF THEN 380
360 FOR X = 2 TO 8 STEP 3:IF B(X) = 0 THEN 520
370 NEXT
380 IF R(6) < > CF THEN 410
390 FOR X = 3 TO 9 STEP 3:IF B(X) = 0 THEN 520
400 NEXT
410 IF R(7) < > CF THEN 440
420 FOR X = 1 TO 9 STEP 4:IF B(X) = 0 THEN 520
430 NEXT
440 IF R(8) < > CF THEN 470
450 FOR X = 3 TO 7 STEP 2:IF B(X) = 0 THEN 520
460 NEXT
470 IF CF < > 18 THEN CF = 18:GOTO 230
480 Y = 0:FOR X = 1 TO 9:IF B(X) = 0 THEN Y = Y + 1:
    C(Y) = X
490 NEXT
500 X = INT (10 * RND (2)):IF X < 1 OR X > Y THEN 50
    0
510 X = C(X)
520 B(X) = 1:GOSUB 690:GOSUB 700:GOTO 130
600 R(1) = B(1) + B(2) + B(3)
610 R(2) = B(4) + B(5) + B(6)
620 R(3) = B(7) + B(8) + B(9)
630 R(4) = B(1) + B(4) + B(7)
640 R(5) = B(2) + B(5) + B(8)
650 R(6) = B(3) + B(6) + B(9)
660 R(7) = B(1) + B(5) + B(9)
670 R(8) = B(3) + B(5) + B(7)
680 RETURN
690 HOME :FOR X = 1 TO 9 STEP 3:HTAB 17:FOR Y = 0 TO
    2:PRINT M$(B(X + Y));" ";:NEXT :PRINT :NEXT :RE
    TURN

```

```

700   FOR X = 1 TO 9:IF B(X) < > 0 THEN NEXT : GOTO 8
      00
710   GOSUB 600
720   FOR X = 1 TO 8
730   IF R(X) = 27 THEN 760
740   IF R(X) = 3 THEN 780
750   NEXT :RETURN
760   GOSUB 690
770   VTAB 20:PRINT "YOU WON.....THAT WAS A GOOD GAME"
      : GOTO 820
780   GOSUB 690
790   VTAB 20:PRINT "I WON.....JUST CAN'T KEEP UP, C
      AN YOU?": GOTO 820
800   GOSUB 690
810   VTAB 20:PRINT "WELL, CHALK ONE UP FOR THE CAT...
      .."
820   PRINT :PRINT "PLAY AGAIN (Y/N): ";
830   GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 83
      0
840   PRINT CH$:POKE 34,0:HOME :IF CH$ < > "Y" THEN P
      RINT :PRINT CHR$ (4);"RUN MENU"
850   RUN

```

10. Qubic

If you think *Tic-Tac-Toe* is too easy, why not try something with a little more "depth?" Qubic is three-dimensional tic-tac-toe. You win by getting three in a row in any direction on any horizontal, vertical, or diagonal plane.

The computer, as your opponent, plays an aggressive game. If you can win playing against the computer, you are ready for the US Olympic Qubic Team. Tryouts are in Kenosha Falls, Wisconsin, on the 5th Monday in February each year.

Play begins with the traditional toss of the chip. You have a 50-50 chance of winning the toss, but the computer does too. The player who wins the toss moves first. From this point on it's every player for himself.

Remember, you or the computer can win with three markers in a row in any direction! There may be more directions than you are first aware of because of the three-dimensional board, so stay alert.

The game's logic is similar to that used in *Tic-Tac-Toe*, except pointer arrays are used to determine the analytical course the program will follow. There are 49 possible ways to win in *Qubic*, thus the amount of analysis that is completed before the computer chooses a move can be formidable. The computer, however, does it with one equation.

Game Rules

1. If you are selecting the game from the Main Menu, choose 1Ø. If you are not using the menu, run the program.
2. Select X's or O's by pressing the X or O key.
3. The computer will randomly select who goes first.
4. Enter the level and position where you want to place your marker.
5. Levels are numbered 1 through 3, from left to right.
6. Positions are numbered, on each level, as the positions in *Tic-Tac-Toe*:

1	2	3
4	5	6
7	8	9

7. The object of the game is to get three of your markers in a row, in any direction.

8. If neither the player nor the computer gets three in a row, the "cat" is the winner.
9. At the completion of each game you can play again or exit to the Main Menu.
10. If you choose to exit, a scorecard is displayed showing how many games were won by each side.

Programming Notes

10-90	Program initialization
100-110	You select type of marker
120	Assignment of X's and O's to you and the computer
130-140	Choose and print who moves first
150-160	Delay to read messages
170	Setup the screen
180	If you go first, display board and get move
190	Computer takes best position, if available
200-210	Get level for move and check validity
220-240	Get position for move and check validity
250-260	Check if position is taken and make move if it is not taken
270	Display new board
280	Check if winner
290-440	Determine the computer's move
450	Display new board and check for winner
500	Routine to display board status
600-660	Routine to check for winner
670-690	You win, display notice
700-720	Computer wins, display notice
730-740	Cat wins, display notice
750-760	Play again or exit to Main Menu
770	End program by showing final score and run the Main Menu program
780-790	Restart game
800-840	Routine to accumulate data arrays
850-950	Data arrays for possible win positions

Qubic.

```

10  REM          QUBIC
20  POKE 750,0:POKE 751,0:POKE 752,0
30  DIM B(3,9),R(49),P(49,3,2),T(27,2)
40  IW = 0: CW = 0: YW = 0
50  FOR X = 1 TO 49: FOR Y = 1 TO 3: READ A: P(X,Y,1) =
    INT (A / 10): P(X,Y,2) = A - P(X,Y,1) * 10: NEXT
    : NEXT
60  M$(0) = ".": M1$ = "CHECKING FOR WINNER": M2$ = "I
    'M THINKING"
70  TEXT : HOME
80  FOR L = 1 TO 3: FOR P = 1 TO 9: B(L,P) = 0: NEXT : N
    EXT
90  PRINT TAB( 18); "QUBIC"
100 PRINT : PRINT "WOULD YOU LIKE X'S OR O'S: ";
110 GET M$: IF M$ < > "X" AND M$ < > "O" THEN 110
120 PRINT M$: M$(9) = M$: M$(1) = "X": IF M$ = "X" THEN
    M$(1) = "O"
130 X = RND (2): IF X > .5 THEN PRINT : PRINT : PRINT "
    I WIN THE TOSS....I GO FIRST 1": GOTO 150
140 PRINT : PRINT "YOU WON THE TOSS OF THE CHIP.....
    YOU GO FIRST 1"
150 FOR DE = 1 TO 1200
160 NEXT DE
170 HOME : PRINT "LEVELS ARE 1, 2, AND 3, LEFT TO RIG
    HT.": PRINT "POSITIONS ARE NUMBERED (ON EACH LEVE
    L) 1 THROUGH 9 FROM THE TOP-LEFT TO THE BOTT
    OM-RIGHT CORNER.": VTAB 15: POKE 34,5: PRINT : HOME
180 IF X < = .5 THEN GOSUB 500: GOTO 200
190 B(2,5) = 1: GOSUB 500
200 VTAB 20: HTAB 1: CALL - 958: PRINT "LEVEL (1-3): "
    ;
210 GET CH$: L = VAL (CH$): IF L < 1 OR L > 3 THEN 210
220 PRINT L: PRINT "POSITION (1-9): ";
230 GET CH$: P = VAL (CH$): IF P < 1 OR P > 9 THEN 230
240 PRINT P
250 IF B(L,P) < > 0 THEN PRINT CHR$( 7): GOTO 200
260 B(L,P) = 9
270 GOSUB 500
280 GOSUB 600: REM SEE IF WINNER
290 REM
300 REM
310 REM SEE IF COMP CAN WIN
320 REM
330 IF B(2,5) = 0 THEN B(2,5) = 1: GOTO 450
340 CF = 2
350 FOR X = 1 TO 49: IF R(X) < > CF THEN 390
360 FOR Y = 1 TO 3: L = P(X,Y,1): P = P(X,Y,2)
370 IF B(L,P) = 0 THEN B(L,P) = 1: GOTO 450
380 NEXT Y
390 NEXT X
400 IF CF < > 18 THEN CF = 18: GOTO 350
410 C = 0: FOR L = 1 TO 3: FOR P = 1 TO 9: IF B(L,P) =
    0 THEN C = C + 1: T(C,1) = L: T(C,2) = P

```

```

420 NEXT :NEXT
430 T = INT (RND (2) * 27):IF T < 1 OR T > C THEN 43
Ø
440 B(T(T,1),T(T,2)) = 1
450 GOSUB 500:GOSUB 600: GOTO 200
500 HOME :FOR L = 1 TO 3:FOR P = 1 TO 7 STEP 3:VTAB
(11 + INT (P / 3)):HTAB (L * 8):FOR Y = P TO P +
2:PRINT M$(B(L,Y));" ";:NEXT :NEXT :NEXT :PRINT
:RETURN
600 VTAB 20:HTAB 1:CALL - 958:T$ = M1$:M1$ = M2$:M2
$ = T$:PRINT M1$;:FOR L = 1 TO 3:FOR P = 1 TO 9:
IF B(L,P) < > Ø THEN NEXT :NEXT : GOTO 730
GOSUB 800
610 FOR X = 1 TO 49
620 IF R(X) = 27 THEN POP : GOTO 670
640 IF R(X) = 3 THEN POP : GOTO 700
650 NEXT
660 RETURN
670 VTAB 20:HTAB 1:CALL - 958:FOR Y = 1 TO 3:PRINT
("";P(X,Y,1);"-";P(X,Y,2);") ";:NEXT :PRINT
680 YW = PEEK (750):YW = YW + 1:POKE 750,YW
690 PRINT "YOU WON.....THAT WAS A GOOD GAME": GOTO 7
50
700 VTAB 20:HTAB 1:CALL - 958:FOR Y = 1 TO 3:PRINT
("";P(X,Y,1);"-";P(X,Y,2);") ";:NEXT :PRINT
710 IW = PEEK (751):IW = IW + 1:POKE 751,IW
720 PRINT "I WON.....JUST CAN'T KEEP UP, CAN YOU ?":
GOTO 750
730 CW = PEEK (752):CW = CW + 1:POKE 752,CW
740 VTAB 20:HTAB 1:CALL - 958:PRINT "WELL, CHALK ON
E UP FOR THE CAT....."
750 PRINT :PRINT "PLAY AGAIN (Y/N): ";
760 GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 76
Ø
770 PRINT CH$:IF CH$ = "N" THEN POKE 34,Ø:HOME :PRIN
T :PRINT :PRINT "YOU WON: ";PEEK (750):PRINT "I
WON: ";PEEK (751):PRINT "CAT WON: ";PEEK (752)
:PRINT :PRINT "THANKS FOR THE GAMES!":PRINT CHR$(
4);"RUN MENU"
780 POKE 34,Ø
790 GOTO 60
800 FOR X = 1 TO 49:R(X) = Ø:IF INT (X / 3) * 3 = X
THEN PRINT ". ";
810 FOR Y = 1 TO 3
820 R(X) = R(X) + B(P(X,Y,1),P(X,Y,2))
830 NEXT Y:NEXT X
840 RETURN
850 DATA 11,12,13,14,15,16,17,18,19,11,14,17,12,15,
18,13,16,19,11,15,19,13,15,17
860 DATA 21,22,23,24,25,26,27,28,29,21,24,27,22,25,
28,23,26,29,21,25,29,23,25,27
870 DATA 31,32,33,34,35,36,37,38,39,31,34,37,32,35,
38,33,36,39,31,35,39,33,35,37
880 DATA 11,21,31,12,22,32,13,23,33
890 DATA 14,24,34,15,25,35,16,26,36

```

Sams Books cover a wide range of technical topics. We are always looking for more information from you, our readers, as to which additional topics need coverage. Please fill out this questionnaire and return it to us with your suggestions. They will be appreciated.

Please check the areas of interest:

1. CURRENT TECHNOLOGIES

- Electronics
- Circuit Design
- Computers
 - Business Applications
 - Fundamentals
 - Languages _____
Specify _____
 - Machine Specific: _____

- Microprocessors
- Networking
- Servicing/Repair
- Other _____

2. NEW TECHNOLOGIES

- Fiber Optics
- Robotics
- Security Electronics
- Speech Synthesis
- Telecommunications
 - Cellular
 - Satellite
- Video
- Other _____

3. Do you own operate a personal computer? Model _____

4. Have you bought other Sams Books? Please list: _____

5. OCCUPATION

- Business Professional _____
Specify _____
- Educator
- Engineer _____
Specify _____
- Hobbyist
- Programmer
- Retailer
- Student
- Other _____

6. EDUCATION

- High School Graduate
- Tech School Graduate
- College Graduate
- Post-graduate degree

COMMENTS _____

(OPTIONAL)

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

22166 22331 22337 22382 22394

Book Markkary

SAMSTM

Book MarkkryarW Book

SAMSTM



BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 1076 INDIANAPOLIS, IND.

POSTAGE WILL BE PAID BY ADDRESSEE

HOWARD W. SAMS & CO., INC.
4300 WEST 62ND STREET
P.O. Box 7092
Indianapolis, IN 46206

ATTENTION: Public Relations Department

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



900 DATA 17,27,37,18,28,38,19,29,39
910 DATA 11,22,33,14,25,36,17,28,39
920 DATA 13,22,31,16,25,34,19,28,37
930 DATA 11,24,37,12,25,38,13,26,39
940 DATA 17,24,31,18,25,32,19,26,33
950 DATA 11,25,39,13,25,37,17,25,33,19,25,31

11. Depth Charge

In *Depth Charge*, you control the nuclear-powered boat *Nemesis*, the fastest boat in the Navy, on a search-and-destroy mission in the North Atlantic. There are always three submarines beneath you, moving at different speeds. You destroy them by hitting them with depth charges. Timing is of the essence and accuracy is a must.

The final score is based on how many submarines are destroyed. The deeper and faster the submarine, the more points you accumulate for destroying it. You only have five passes across the ocean, so don't hesitate.

This program uses high-resolution shape tables to create the illusion of motion. Because it is possible to have up to nine items on the screen at one time (one ship, three submarines, and five depth charges), there are times when the computer is very "busy" redrawing shapes from the predefined tables. When this happens, the game's speed slows down. Do not despair, however, as the pace increases as the number of items on the screen decreases.

Play the game and study the listing to learn more about using shape tables. All the shapes in this game (and in other programs in *Apple Games*) were made using the *Shape Table Generator* program (see Program 12).

Game Rules

1. If you are selecting the game from the Main Menu, choose 11. If you are not using the menu, run the program.
2. The instruction screen will be displayed.
3. Press <RETURN> to begin play or <ESC> to exit the program.
4. Once the game begins, press any key to drop depth charges.
5. Up to five depth charges can be on the screen at one time.
6. The object of the game is to destroy as many enemy submarines as possible by hitting them with depth charges.
7. As each submarine is destroyed, it is replaced with another one.
8. The game is over when you have crossed the screen the allotted number of times.

9. When the game is completed, the program returns to the instruction screen to play again or exit to the Main Menu.

Programming Notes

1Ø-4Ø	Program initialization
5Ø-6Ø	Load and initialize shape table
7Ø	Branch to instruction screen
1ØØ	Game initialization
11Ø	Generate locations for the three submarines
12Ø	Draw initial ship and pass counter
13Ø	Draw initial submarines
2ØØ-21Ø	Draw ship, calculating new position. Increment counters
22Ø	Check for keypress
23Ø-26Ø	A keypress, so drop depth charge if less than five depth charges are displayed
27Ø	Short delay if no depth charges are on the screen
28Ø-31Ø	Draw depth charges
32Ø-36Ø	Check if depth charge has hit a submarine and blow up the submarine if it is a hit
37Ø-43Ø	Check if depth charges are near the bottom of the screen; if they are, remove them
44Ø-46Ø	Draw submarines in correct places and calculate their next position
47Ø	Branch again
5ØØ-53Ø	Determine location for submarines
55Ø-63Ø	Blow up submarine and adjust score
64Ø-69Ø	Initial playing screen, draw border and sea
7ØØ-75Ø	Initialize game pointers
8ØØ-84Ø	Display instruction screen
85Ø-88Ø	Get keypress and determine action

Depth Charge.

```

1Ø  REM          DEPTH CHARGE
2Ø  M1 = PEEK (115):M2 = PEEK (116)
3Ø  HIMEM: 8ØØØ
4Ø  DIM SP(3,3),DC(5,2)
5Ø  PRINT "BLOAD SUB.OBJ"
6Ø  POKE 232,32:POKE 233,78
7Ø  GOTO 8ØØ
1ØØ HOME :GOSUB 7ØØ:GOSUB 65Ø
11Ø FOR X = 1 TO 3:GOSUB 5ØØ:NEXT

```

```

120 SH = 275:XDRAW 1AT SH,19:PA = 1:VTAB 23:HTAB 31:
PRINT "PASS: ";PA
130 FOR X = 1 TO 3:XDRAW 2AT SP(X,1),SP(X,2):NEXT
200 XDRAW 1AT SH,19:SH = SH - 3:IF SH < = 4 THEN SH
= 275:PA = PA + 1:VTAB 23:HTAB 31:PRINT "PASS:
";PA
210 XDRAW 1AT SH,19:IF PA > NP THEN 800
220 KB = PEEK ( - 16384):IF KB < 128 THEN 270
230 POKE - 16368,0:REM RESET STROBE
240 IF DC > = 5 THEN 270
250 DC = DC + 1:DC(DC,1) = SH:DC(DC,2) = 21
260 XDRAW 3AT DC(DC,1),DC(DC,2)
270 IF DC = 0 THEN FOR J = 1 TO 300:NEXT : GOTO 370
280 FOR J = 1 TO DC
290 XDRAW 3AT DC(J,1),DC(J,2)
300 DC(J,2) = DC(J,2) + 3:IF DC(J,2) > = 157 THEN T
8 = 1: GOTO 360
310 XDRAW 3AT DC(J,1),DC(J,2)
320 FOR X = 1 TO 3:IF DC(J,1) < SP(X,1) - 5 OR DC(J,
1) > SP(X,1) + 6 THEN 350
330 IF DC(J,2) < SP(X,2) - 6 OR DC(J,2) > SP(X,2) TH
EN 350
340 GOSUB 550:GOSUB 500:SP(X,1) = 1:XDRAW 2AT SP(X,1
),SP(X,2)
350 NEXT
360 NEXT
370 T9 = 0:IF DC = 0 OR T8 = 0 THEN 440
380 FOR J = 1 TO DC
390 IF DC(J,2) > = 157 THEN MI = MI + 1: GOTO 410
400 T9 = T9 + 1:DC(T9,1) = DC(J,1):DC(T9,2) = DC(J,2
)
410 NEXT
420 DC = T9
430 HTAB 1:VTAB 23:PRINT "MISSES: ";MI - HI:HTAB 1:V
TAB 22:PRINT "HITS: ";HI
440 T8 = 0
450 FOR X = 1 TO 3:XDRAW 2AT SP(X,1),SP(X,2):SP(X,1)
= SP(X,1) + SP(X,3):IF SP(X,1) > = 275 THEN SP
(X,1) = 4
460 XDRAW 2AT SP(X,1),SP(X,2):NEXT
470 GOTO 200
500 T9 = INT (RND (1) * 10):IF T9 < 1 THEN T9 = 1
510 SP(X,3) = T9:SP(X,1) = INT (RND (1) * 200)
520 SP(X,2) = INT (RND (1) * 155):IF SP(X,2) < 25 OR
SP(X,2) > 155 THEN 520
530 RETURN
550 FOR Q = 2 TO 4:SCALE= Q:XDRAW 2AT SP(X,1),SP(X,2
):VTAB 1:HTAB 1:PRINT "":NEXT
560 FOR Q = 4 TO 1 STEP - 1:SCALE= Q:XDRAW 2AT SP(X
,1),SP(X,2):NEXT
570 TT = TT + INT (SP(X,2) / 2 + SP(X,3) * 5)
580 XDRAW 3AT DC(J,1),DC(J,2):DC(J,2) = 160
590 HI = HI + 1
600 IF TT > HT THEN HT = TT
610 IF TT > 500 THEN NP = 6
620 IF TT > 750 THEN NP = 7

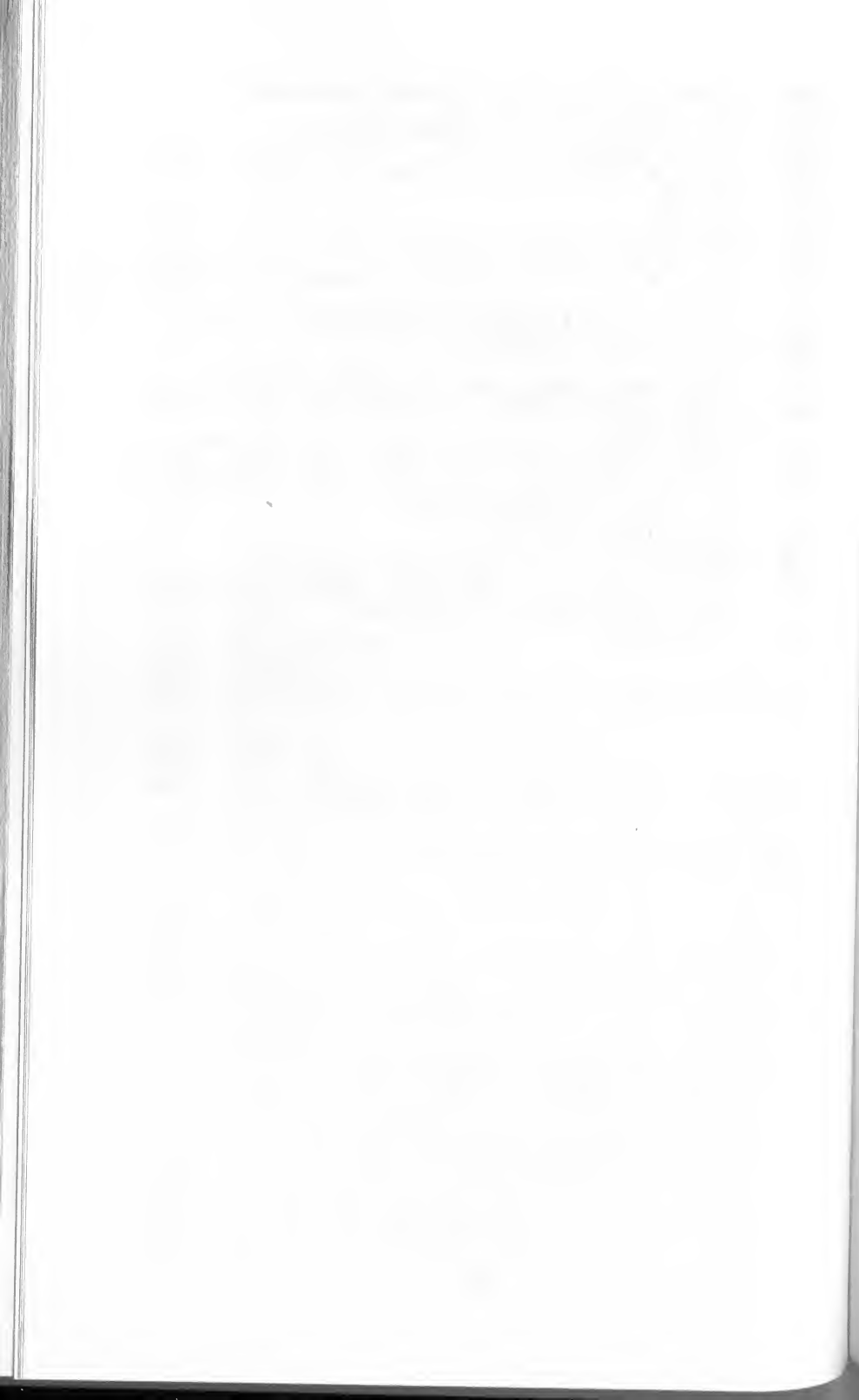
```



```

630 HTAB 30:VTAB 22:PRINT "SCORE: ";TT:RETURN
640 REM DRAW BORDER AND SEA
650 HGR :HCOLOR= 3:HPLLOT 0,20 TO 279,20
660 ROT= 0:SCALE= 1
670 HPLLOT 0,0 TO 279,0 TO 279,159 TO 0,159 TO 0,0
680 TT = 0
690 RETURN
700 BB = 159:LB = 0:RB = 279:SL = 20
710 NP = 5:TT = 0:MI = 0:HI = 0
720 FOR X = 1 TO 3:FOR Y = 1 TO 3:SP(X,Y) = 0:NEXT :
NEXT
730 FOR X = 1 TO 5:DC(X,1) = 0:DC(X,2) = 0:NEXT
740 DC = 0:POKE - 16368,0
750 RETURN
800 TEXT :HOME :PRINT TAB( 13);"DEPTH CHARGES"
810 PRINT :PRINT "SCORE: ";TT;TAB( 20);"HIGH SCORE:
";HT:PRINT
820 PRINT "PRESS <RETURN> TO PLAY, <ESC> TO END"
830 PRINT :PRINT :PRINT "ANY KEY FIRES DEPTH CHARGES
- ONLY FIVE PASSES ALLOWED - EXTRA PASSES AT 50
0 AND 750 POINTS":PRINT
840 VTAB 5:HTAB 38:PRINT " ";
850 GET A$
860 IF A$ = CHR$ (13) THEN PRINT : GOTO 100
870 IF A$ = CHR$ (27) THEN PRINT :POKE 115,M1:POKE 1
16,M2:PRINT CHR$ (4);"RUN MENU"
880 GOTO 850

```



Section 3
UTILITY AND MISCELLANEOUS
PROGRAMS

12. Shape Table Generator

This utility allows you to quickly and easily create high-resolution shape tables for use in your programs. While the program is in use, note that the actual design is produced in low-resolution graphics. This is so you can easily see and create the details of your design.

Several programs in *Apple Games* use high-resolution graphics shape tables. All of these tables were created using this program. With a little practice, you should be able to easily create similar designs and shape tables.

Program Instructions

1. If you are selecting the program from the Main Menu, choose 12. If you are not using the menu, run the program.
2. The first question asked is the beginning address for the shape table. Enter a decimal number representing where your shape table will be stored in memory. This can be any location as long as it will not be "disturbed" by any other program or variables the computer has in memory at the time it is later used. I use a large number, such as 20000, and this works well in most instances.
3. The next question is how many shapes you will be making for the table. This should be a decimal number from 1 to 255. If you use more than 255 shapes, your shape table will not work properly.
4. Finally, you will be asked the name of your shape table. This can be any valid DOS (Disk Operating System) file name. It must begin with a letter and it cannot contain commas or colons. Total length for the file name should not exceed 29 characters.
5. The drawing screen will be displayed. The status lines are at the bottom of the screen. Note the direction and mode indicators.
6. To change the cursor's direction, use the I, J, K, and M keys. These represent up, left, right, and down, respectively. If you are using an Apple IIe or Apple IIc, you can use the arrow keys to change the direction of the cursor. Try these keys and note their effect on the indicators.

7. The mode can be in one of two states — Plot or NoPlot. To change the mode, use the X key to toggle between the two states. Try this and note its effect on the mode indicator.
8. Now you are ready to draw your shape. Be sure you are in Plot mode and choose a direction. Press the <SPACE BAR> and the plotting cursor moves in the direction you have chosen. Change mode and direction to construct various designs.
9. *Warning:* Do not move the cursor more than two spaces UP when in the NoPlot mode. If you do, a portion of the shape or shape table will be lost. This is a limitation of Applesoft. To get around this limitation, you can go up two spaces, left one space, right one space, then up two more spaces, and so on.
10. To start over, press C to clear the screen.
11. To start your shape at any specific X,Y coordinate, press R. If you do not specify a starting location, the center of the screen is used.
12. When you have completed a shape, press <ESC> and the computer will save it to memory. This process may take a few moments, depending on the shape's complexity. You can then draw another shape for your table.

Programming Notes

It is easy to use your shape table in programs. The first step is to BLOAD the table into memory. Then POKE the pointers to the beginning of the table into memory locations 232 and 233. The following Applesoft BASIC instructions will accomplish this task:

```
X = Beginning decimal address of table
J = INT ( X / 256 ) : K = X - J * 256
POKE 232 , K : POKE 233 , J
```

Now you can use your shape table with commands such as DRAW and XDRAW. These commands are detailed in the *Applesoft Reference Manual*.

Experiment and have fun. Be sure the file name you specify is not the same as another file on the disk, or you could erase the original file.

10-40	Program initialization
50	Get and validate beginning address for table
60	Get and validate number of shapes
70	POKE values for number of shapes

80-90	Get and validate name for shape table
100-140	Program initialization
150-350	Translate shape and POKE into memory
360-370	Query for another shape
380	Increment shape counter, see if table is full
390-480	Set up plotting screen and initial modes
490	Get keypress and convert it to uppercase
500	Make "click" to show a keypress
510-590	Check for valid keypresses
600	Keypress error, ring bell and loop back
610-620	Switch between Plot and NoPlot modes
630-650	Command to clear shape
660	Change direction to up
670	Change direction to down
680	Change direction to left
690	Change direction to right
700-810	Command to move cursor
820-880	Command to reset plotter
890-930	Subroutine for shape-translation process
940-960	Get Y/N response and convert it to uppercase
970-1080	Short instruction display
1090-1110	Prompt and get keypress
1120-1150	Save shape table to disk and exit to the Main Menu
1160-1290	REMark statements explaining how to use tables

Shape Table Generator.

```

10 REM SHAPE TABLE GENERATOR
20 M1 = PEEK (115):M2 = PEEK (116):HIMEM: 8192
30 GOSUB 970
40 DEF FN MOD(Q) = Q - INT (Q / 256) * 256
50 INPUT "WHAT IS THE BEGINNING (DECIMAL) ADDRESS O
F THE TABLE? ";A$:BA = INT (VAL (A$)):IF BA < 25
6 OR BA > 65536 THEN 50
60 INPUT "HOW MANY SHAPES ? ";N$:N = INT (VAL (N$))
:IF N < 1 THEN 60
70 POKE BA,N:POKE BA + 1,0
80 INPUT "NAME OF TABLE ? ";N9$
90 IF LEN (N9$) = 0 THEN 80
100 SN = 1:A = BA + 2 + (N * 2)
110 POKE 233,INT (BA / 256)
120 POKE 232,FN MOD(BA)
130 DIM A(500)
140 GOTO 400
150 A(MA) = 255
160 A1 = A
170 TR = 0:D1 = 0
180 FOR X = 0 TO MAX

```

```

190 C = A(X)
200 IF C = 0 THEN 890
210 IF C = 255 THEN 310
220 IF TR + 8 ^ D1 * C > 255 THEN 270
230 TR = TR + 8 ^ D1 * C
240 D1 = D1 + 1
250 NEXT
260 STOP
270 POKE A,TR
280 A = A + 1
290 TR = 0:D1 = 0
300 GOTO 220
310 POKE A,TR:A = A + 1:POKE A,0
320 A = A + 1:POKE A,0
330 L = A - A1
340 POKE BA + (SN * 2) + 1,INT ((A - BA - L) / 256)
350 POKE BA + (SN * 2),FN MOD(A - BA - L)
360 VTAB 23:CALL - 958:PRINT "ANOTHER SHAPE (Y/N):
";:GOSUB 940
370 IF A$ = "N" THEN 1120
380 SN = SN + 1:IF SN > N THEN SN = SN - 1:PRINT "SO
RRY, SHAPE TABLE IS FULL": GOTO 1120
390 REM ENTER SHAPE
400 GR
410 VTAB 22:PRINT "SHAPE NUMBER: ";SN
420 MA = 0:L = 0
430 COLOR= 15
440 X = 20:Y = 20
450 REM SET UP INITIAL MODES
460 C$ = "MODE: PLOT":D$ = "UP " :C = 1:D = 0
470 O = SCRNB( X,Y):COLOR= 2:PLOT X,Y
480 VTAB 23:CALL - 958:VTAB 23:PRINT C$;" DIREC
TION: ";D$;:HTAB 1
490 GET A$:IF ASC (A$) > 96 AND ASC (A$) < 123 THEN
A$ = CHR$ (ASC (A$) - 32)
500 F = PEEK ( - 16336) + PEEK ( - 16336)
510 IF A$ = CHR$ (27) THEN 150
520 IF A$ = "X" THEN 610
530 IF A$ = "C" THEN 630
540 IF A$ = "I" OR A$ = CHR$ (11) THEN 660
550 IF A$ = "M" OR A$ = CHR$ (10) THEN 670
560 IF A$ = "J" OR A$ = CHR$ (8) THEN 680
570 IF A$ = "K" OR A$ = CHR$ (21) THEN 690
580 IF A$ = " " THEN 700
590 IF A$ = "R" THEN 820
600 PRINT CHR$ (7);: GOTO 480
610 C = NOT C:C$ = "PLOT " :IF NOT C THEN C$ = "NOPL
OT"
620 C$ = "MODE: " + C$: GOTO 480
630 VTAB 23:CALL - 958:PRINT "CLEAR? (Y/N): ";:GOSU
B 940
640 IF A$ = "N" THEN HTAB 1: GOTO 480
650 GOTO 400
660 D$ = "UP " :D = 0: GOTO 480
670 D$ = "DOWN " :D = 2: GOTO 480

```

```

680 D$ = "LEFT ":D = 3: GOTO 480
690 D$ = "RIGHT":D = 1: GOTO 480
700 IF D = 0 AND Y = 0 THEN 470
710 IF D = 1 AND X = 39 THEN 470
720 IF D = 2 AND Y = 39 THEN 470
730 IF D = 3 AND X = 0 THEN 470
740 COLOR= 0:PLOT X,Y:COLOR= 15:IF C THEN PLOT X,Y
750 ON D GOTO 770,780,790
760 Y = Y - 1: GOTO 800
770 X = X + 1: GOTO 800
780 Y = Y + 1: GOTO 800
790 X = X - 1
800 TY = 4 * C + D:A(MA) = TY:MA = MA + 1
810 GOTO 470
820 VTAB 23:CALL - 958:PRINT "RESET? (Y/N): ";:GOSUB
  B 940
830 IF A$ = "N" THEN HTAB 1: GOTO 480
840 GR
850 VTAB 23:CALL - 958:INPUT "ENTER X: ";A$:X = VAL
  (A$):IF X < 0 OR X > 39 THEN 850
860 VTAB 23:CALL - 958:INPUT "ENTER Y: ";A$:Y = VAL
  (A$):IF Y < 0 OR Y > 39 THEN 860
870 MAX = 0
880 GOTO 450
890 IF D1 = 0 THEN 210
900 IF A(X + 1) < 4 AND A(X + 1) < > 0 THEN 210
910 IF A(X + 1) = 0 THEN D1 = 2:X = X + 1: GOTO 930
920 D1 = 1
930 POKE A,TR:A = A + 1:TR = 0: GOTO 250
940 GET A$:IF ASC (A$) > 96 AND ASC (A$) < 123 THEN
  A$ = CHR$ (ASC (A$) - 32)
  IF A$ < > "Y" AND A$ < > "N" THEN 940
950 PRINT A$:RETURN
970 TEXT :HOME
980 PRINT "SHAPE TABLE GENERATOR WILL HELP YOU DRAWH
  I-RESOLUTION SHAPE TABLES FOR USE IN YOUR PROG
  RAMS."
990 PRINT
1000 PRINT "THE KEYS I, J, K, AND M CONTROL DIREC- T
  ION AND ARE USED FOR UP, LEFT, RIGHT, AND DOWN
  RESPECTIVELY. THE ARROW KEYS MAY ALSO BE USED,
  IF DESIRED."
1010 PRINT
1020 PRINT "THE SPACE BAR WILL MOVE THE CURSOR IN T
  HE DIRECTION DESIRED. TO CHANGE FROM"
1030 PRINT "PLOT TO NO-PLOT USE THE X KEY. THE C"
1040 PRINT "KEY WILL CLEAR THE SHAPE SO YOU MAY"
1050 PRINT "START OVER. R WILL RESET THE COORDINATES"
  ;
1060 PRINT "TO ALLOW YOU TO BEGIN THE SHAPE AT A":PRI
  NT "POINT OTHER THAN THE CENTER."
1070 PRINT
1080 PRINT "RESPOND TO ALL QUESTIONS (Y/N)":PRINT
1090 PRINT "PRESS <RETURN> TO CONTINUE...";
1100 GET A$:IF A$ < > CHR$ (13) THEN 1100

```



```
1110 PRINT :HOME :RETURN
1120 PRINT "BSAVE";N9$;"A";BA;"L";A - BA + 2
1130 TEXT :HOME
1140 POKE 115,M1:POKE 116,M2
1150 PRINT CHR$(4)"RUN MENU"
1160 REM
1170 REM
1180 REM WHEN USING THE TABLE
1190 REM AT A LATER DATE, SET
1200 REM THE BEGINNING ADDRESS
1210 REM OF THE SHAPE TABLE IN
1220 REM MEMORY LOCATIONS $E8
1230 REM & $E9, LOW BYTE FIRST
1240 REM
1250 REM ALSO, IT IS A GOOD IDEA
1260 REM TO SET HIMEM BELOW
1270 REM THE TABLE'S STARTING
1280 REM ADDRESS TO PROTECT IT
1290 REM
```

13. Opening Ceremonies

Welcome to the welcome!

This program is the high-resolution demonstration that spells APPLE GAMES. This is the second screen displayed when you boot the *Apple Games* disk.

After studying the program listing, you may discover how to do a few tricks with your Apple. *Opening Ceremonies* uses a small shape table and the HPLLOT function to create the "sign-board" effect that is displayed when the program is run. This same effect can be used (with modifications) to display virtually any message.

Program Instructions

1. If you are selecting the game from the Main Menu, choose 13. If you are not using the menu, run the program.
2. The demonstration screen will be displayed.
3. Press <ESC> to exit to the Main Menu.

Programming Notes

10-30	Load and initialize shape table
40-50	Initialize display screen
60	Fill top half of screen with color
70-90	Randomly choose letter order
100	Initialize pointers
110-120	Choose letter to display next
130	Set color
140	Check for keypress
150-170	Finish displaying letters
180-240	Execute special effects on screen
250-260	Loop and repeat the demonstration screen
270	Position and print letter A
280	Position and print letter P
290	Position and print letter P
300	Position and print letter L
310	Position and print letter E
320	Position and print letter G
330	Position and print letter A

34Ø	Position and print letter M
35Ø	Position and print letter E
36Ø	Position and print letter S
37Ø-39Ø	Randomly change directions
40Ø	Routine to draw second shape
41Ø	Routine to draw first shape
42Ø-999	Check for <ESC> being pressed, if so run the Main Menu program
100ØØ-103Ø	Plot letter A
110ØØ-112Ø	Plot letter P
120ØØ	Plot letter L
130ØØ-133Ø	Plot letter E
140ØØ-143Ø	Plot letter G
150ØØ-151Ø	Plot letter M
160ØØ-163Ø	Plot letter S

Opening Ceremonies.

```

1Ø    REM          OPENING CEREMONIES
2Ø    PRINT "BLOAD ONE SHPS"
3Ø    POKE 233,78:POKE 232,32
4Ø    HGR
5Ø    HOME :VTAB 22:PRINT TAB( 1Ø);"PRESS <ESC> FOR ME
      NU":VTAB 1
6Ø    HCOLOR= 3:FOR X = Ø TO 169:H PLOT Ø,X TO 279,X:NE
      XT
7Ø    C = INT (RND (1) * 8):C1 = INT (RND (1) * 8)
8Ø    IF C1 < > Ø AND C1 < > 3 AND C1 < > 4 AND C1
      < > 7 AND C < > Ø AND C < > 3 AND C < > 4 AN
      D C < > 7 THEN 1ØØ
9Ø    IF C1 + 3 = C OR C + 3 = C1 THEN 7Ø
10Ø   FOR X = 1 TO 1Ø:A(X) = Ø:NEXT
11Ø   Z = INT (RND (1) * 15):IF Z < 1 OR Z > 1Ø THEN 1
      1Ø
12Ø   IF A(Z) < > Ø THEN 11Ø
13Ø   HCOLOR= C:IF Z > 5 THEN HCOLOR= C1
14Ø   GOSUB 42Ø
15Ø   A(Z) = 1:ON ZGOSUB 27Ø,28Ø,29Ø,30Ø,31Ø,32Ø,33Ø,3
      4Ø,35Ø,36Ø
16Ø   FOR W = 1 TO 1Ø:IF A(W) = Ø THEN 11Ø
17Ø   NEXT
18Ø   ROT= Ø
19Ø   A = Ø:B = 9Ø:ST = 1:GOSUB 37Ø
20Ø   GOSUB 40Ø
21Ø   A = Ø:B = 279:ST = 1:GOSUB 37Ø
22Ø   GOSUB 41Ø
23Ø   A = Ø:B = 16Ø:ST = - 1:GOSUB 37Ø
24Ø   GOSUB 40Ø
25Ø   GOTO 7Ø
26Ø   END

```

```

270 X = 30:Y = 15:GOSUB 1000:X = 31:Y = 16:GOSUB 100
0:RETURN
280 X = 80:Y = 12:GOSUB 1100:X = 81:Y = 13:GOSUB 110
0:RETURN
290 X = 130:Y = 12:GOSUB 1100:X = 131:Y = 13:GOSUB 1
100:RETURN
300 X = 180:Y = 15:GOSUB 1200:X = 181:Y = 16:GOSUB 1
200:RETURN
310 X = 195:Y = 15:GOSUB 1300:X = 196:Y = 16:GOSUB 1
300:RETURN
320 X = 15:Y = 82:GOSUB 1400:X = 16:Y = 83:GOSUB 140
0:RETURN
330 X = 65:Y = 85:GOSUB 1000:X = 66:Y = 86:GOSUB 100
0:RETURN
340 X = 115:Y = 85:GOSUB 1500:X = 116:Y = 86:GOSUB 1
500:RETURN
350 X = 165:Y = 85:GOSUB 1300:X = 166:Y = 86:GOSUB 1
300:RETURN
360 X = 215:Y = 85:GOSUB 1600:X = 216:Y = 86:GOSUB 1
600:RETURN
370 Q = RND (1) * 10:IF Q > 5 THEN ST = - ST
380 IF ST < > ABS (ST) THEN C = A:A = B:B = C
390 RETURN
400 SCALE= 28:FOR X = A TO B STEP ST:XDRAW 2AT 279,X
:GOSUB 420:NEXT :RETURN
410 SCALE= 17:FOR X = A TO B STEP ST:XDRAW 1AT X,167
:GOSUB 420:NEXT :RETURN
420 P = PEEK ( - 16384):POKE - 16368,0:IF P < > 15
5 THEN RETURN
430 PRINT "RUN MENU"
999 END
1000 HPLOT X + 14,Y + 15 TO X + 39,Y + 15 TO X + 43,Y
+ 17 TO X + 48,Y + 21 TO X + 50,Y + 26 TO X + 5
0,Y + 50 TO X + 14,Y + 50 TO X + 10,Y + 48 TO X
+ 8,Y + 46 TO X + 6,Y + 43 TO X + 5,Y + 39 TO X
+ 5,Y + 37 TO X + 6,Y + 34 TO X + 10,Y + 30 TO X
+ 14,Y + 28 TO X + 40,Y + 28 TO X + 39,Y + 25 T
O X + 38,Y + 24 TO X + 36,Y + 23 TO X + 14,Y + 2
3
1010 HPLOT X + 14,Y + 23 TO X + 11,Y + 21 TO X + 11,Y
+ 17 TO X + 14,Y + 15
1020 HPLOT X + 18,Y + 34 TO X + 40,Y + 34 TO X + 40,Y
+ 40 TO X + 18,Y + 40 TO X + 16,Y + 39 TO X + 1
5,Y + 37 TO X + 15,Y + 36 TO X + 16,Y + 35 TO X
+ 18,Y + 34
1030 RETURN
1100 HPLOT X + 5,Y + 20 TO X + 39,Y + 20 TO X + 43,Y
+ 22 TO X + 46,Y + 24 TO X + 48,Y + 27 TO X + 50
,Y + 31 TO X + 50,Y + 39 TO X + 48,Y + 43 TO X +
46,Y + 46 TO X + 43,Y + 48 TO X + 39,Y + 50 TO
X + 15,Y + 50 TO X + 15,Y + 70 TO X + 5,Y + 70 T
O X + 5,Y + 20
1110 HPLOT X + 15,Y + 30 TO X + 36,Y + 30 TO X + 39,Y
+ 32 TO X + 40,Y + 34 TO X + 40,Y + 36 TO X + 3
9,Y + 38 TO X + 36,Y + 40 TO X + 15,Y + 40 TO X
+ 15,Y + 30

```

1120 RETURN
 1200 H PLOT X + 5, Y - 5 TO X + 15, Y - 5 TO X + 15, Y +
 50 TO X + 5, Y + 50 TO X + 5, Y - 5: RETURN
 1300 H PLOT X + 16, Y + 15 TO X + 39, Y + 15 TO X + 44, Y
 + 17 TO X + 49, Y + 22 TO X + 50, Y + 25 TO X + 5
 0, Y + 28 TO X + 49, Y + 31 TO X + 47, Y + 33 TO X
 + 44, Y + 34 TO X + 39, Y + 35 TO X + 15, Y + 35 TO
 X + 15, Y + 37 TO X + 16, Y + 39 TO X + 19, Y + 40
 TO X + 40, Y + 40 TO X + 45, Y + 42 TO X + 46, Y +
 44 TO X + 46, Y + 46 TO X + 45, Y + 48 TO X + 40,
 Y + 50
 1310 H PLOT X + 40, Y + 50 TO X + 16, Y + 50 TO X + 13, Y
 + 49 TO X + 6, Y + 42 TO X + 5, Y + 40 TO X + 5, Y
 + 25 TO X + 6, Y + 22 TO X + 11, Y + 17 TO X + 13
 , Y + 16 TO X + 16, Y + 15
 1320 H PLOT X + 19, Y + 23 TO X + 38, Y + 23 TO X + 40, Y
 + 25 TO X + 40, Y + 27 TO X + 23, Y + 29 TO X + 1
 5, Y + 29 TO X + 15, Y + 27 TO X + 16, Y + 25 TO X
 + 17, Y + 24 TO X + 19, Y + 23
 1330 RETURN
 1400 H PLOT X + 16, Y + 20 TO X + 50, Y + 20 TO X + 50, Y
 + 59 TO X + 48, Y + 63 TO X + 46, Y + 66 TO X + 4
 3, Y + 68 TO X + 39, Y + 70 TO X + 15, Y + 70 TO X
 + 10, Y + 68 TO X + 9, Y + 66 TO X + 9, Y + 64 TO X
 + 10, Y + 62 TO X + 15, Y + 60 TO X + 36, Y + 60 T
 O X + 38, Y + 59 TO X + 40, Y + 56 TO X + 40, Y + 5
 0 TO X + 16, Y + 50 TO X + 12, Y + 48
 1410 H PLOT X + 12, Y + 48 TO X + 9, Y + 46 TO X + 7, Y +
 43 TO X + 5, Y + 39 TO X + 5, Y + 31 TO X + 7, Y +
 27 TO X + 9, Y + 24 TO X + 12, Y + 22 TO X + 16, Y
 + 20
 1420 H PLOT X + 19, Y + 30 TO X + 40, Y + 30 TO X + 40, Y
 + 40 TO X + 19, Y + 40 TO X + 16, Y + 38 TO X + 1
 5, Y + 36 TO X + 15, Y + 34 TO X + 16, Y + 32 TO X
 + 19, Y + 30
 1430 RETURN
 1500 H PLOT X + 27, Y + 36 TO X + 40, Y + 15 TO X + 50, Y
 + 15 TO X + 50, Y + 50 TO X + 40, Y + 50 TO X + 4
 0, Y + 34 TO X + 32, Y + 50 TO X + 23, Y + 50 TO X
 + 15, Y + 34 TO X + 15, Y + 50 TO X + 5, Y + 50 TO
 X + 5, Y + 15 TO X + 15, Y + 15 TO X + 28, Y + 36
 1510 RETURN
 1600 H PLOT X + 16, Y + 15 TO X + 39, Y + 15 TO X + 44, Y
 + 16 TO X + 45, Y + 18 TO X + 45, Y + 20 TO X + 4
 4, Y + 22 TO X + 39, Y + 24 TO X + 17, Y + 24 TO X
 + 15, Y + 26 TO X + 15, Y + 27 TO X + 17, Y + 28 TO
 X + 40, Y + 28 TO X + 44, Y + 30 TO X + 47, Y + 32
 TO X + 49, Y + 35 TO X + 50, Y + 38 TO X + 50, Y +
 40 TO X + 49, Y + 43 TO X + 47, Y + 46
 1610 H PLOT X + 47, Y + 46 TO X + 44, Y + 48 TO X + 40, Y
 + 50 TO X + 16, Y + 50 TO X + 11, Y + 48 TO X + 1
 0, Y + 46 TO X + 10, Y + 44 TO X + 11, Y + 42 TO X
 + 16, Y + 41 TO X + 38, Y + 41 TO X + 40, Y + 39 TO
 X + 40, Y + 38 TO X + 38, Y + 36 TO X + 16, Y + 36
 TO X + 12, Y + 34 TO X + 9, Y + 32 TO X + 7, Y + 2
 9 TO X + 6, Y + 28 TO X + 6, Y + 24

```
1620 HPLOT X + 6,Y + 24 TO X + 9,Y + 19 TO X + 12,Y +  
17 TO X + 16,Y + 15  
1630 RETURN
```

14. Master Catalog

This is a program that you may find useful. If you are like many computer-enthusiasts, you have a large collection of disks. On those disks are assorted programs and who knows what else. Precisely the point of *Master Catalog*. It will allow you to organize and keep track of the program names so you can find them easily. It is a short data-base management program that acts as a filing system for your disks and programs.

You can search by program name, language and type, record number, date entered (into the system), and disk location (which disk the program is on). You can even sort all the records in your file.

As you use the program, note that many of the routines do special functions, such as allowing string input that includes virtually any character. There is also a routine that allows the analysis and reformatting of entered dates. The sorting routine used is QuickSort. The detailed explanation of these routines is beyond the scope of this book, but information on how they work is in *BASIC Tricks for the Apple* (Howard W. Sams #22208).

Program Instructions

1. If you are selecting the program from the Main Menu, choose 14. If you are not using the menu, run the program.
2. The first requirement is entering the date. Your response should be in the format MM/DD/YY.
3. After you have entered the date, the following menu choices will be displayed:
 - 1) ENTER LISTING
 - 2) CHANGE LISTING
 - 3) DELETE LISTING
 - 4) SEARCH LISTING
 - 5) PRINT LISTING
 - 6) SORT LISTINGS
 - 7) EXIT PROGRAM
4. To do any of the functions listed on the menu, press the number of that function.

5. Each function is self-prompting; follow the directions and answer the questions as they appear on the screen.
6. ENTER LISTING will add a record to the system. A record consists of the following information:

PROGRAM NAME The file's name as it appears in the disk catalog. It can be up to 35 characters long.

LANGUAGE TYPE Enter the language type for the file, if applicable.

PROGRAM TYPE Choose from the types shown on the screen. They are:

- 1) **Game**
- 2) **Business**
- 3) **Utility**
- 4) **Text File**
- 5) **Education**
- 6) **Other**

DATE ENTERED This is the system date you entered when you first started the program.

DISK LOCATION This is any location that helps you locate the disk at a later time. This field allows a maximum of five characters.

7. CHANGE LISTING allows you to change any information that was entered through the ENTER LISTING function.
8. DELETE LISTING allows individual records to be deleted from the data base. You will need to enter the record number of the record to be deleted. Then the record is displayed and you are queried as to whether you want to delete the record shown.
9. SEARCH LISTING searches the file for a specific record or for all records that match a "keyword" that you enter. To search for a match, specify the record to recall or the field to use.
10. PRINT LISTING displays all records. There is a pause after each screen of information. Press <RETURN> to proceed to the next screen.
11. SORT LISTINGS sorts the records in the data base by any specific field. Sorting is done in ascending order.

12. EXIT PROGRAM saves your modified data base to disk and then runs the Main Menu.

Programming Notes

10-20	Program initialization
30-100	Open and read program file from disk
110-130	Program initialization
200-240	String input routine
250-305	Date analysis routine
325-330	Get a Y/N response
350	Print underline characters for input field
400	Invalid record number error message
500-520	Get and analyze date
600-710	Print program menu
720-730	Prompt for choice and validate response
740	Branch to appropriate program section
1000-1160	Routine for inputting a record
2000-2260	Routine for changing a record
3000-3150	Routine for deleting a record
4000-4400	Routine for searching records
5000-5130	Routine for displaying records
6000-6210	Routine for sorting records
7000-7090	Save file, exit program, and transfer to the Main Menu

Master Catalog.

```

10  REM          MASTER CATALOG
20  D$ = CHR$(4):OP$ = D$ + "OPEN":CL$ = D$ + "CLOSE":RD$ = D$ + "READ":WR$ = D$ + "WRITE"
30  ONERR GOTO 90
40  PRINT D$;"UNLOCK PROGRAMS"
50  PRINT OP$;"PROGRAMS":PRINT RD$;"PROGRAMS"
60  INPUT N:DIM T$(N + 50,5):IF N = 0 THEN 100
70  FOR X = 1 TO N:FOR Y = 1 TO 5
80  MX = 35:GOSUB 200:T$(X,Y) = I$:NEXT :NEXT : GOTO 100
90  N = 0
100 PRINT CL$:POKE 216,0
110 PT$(1) = "GAME":PT$(2) = "BUSINESS":PT$(3) = "UTILITY":PT$(4) = "TEXT FILE":PT$(5) = "EDUCATION":PT$(6) = "OTHER"
120 NT = 6
130 GOTO 500
200 A$ = "":I$ = ""
205 GET A$:A$ = LEFT$(A$,1):A = ASC (A$)
210 IF A = 13 THEN PRINT :RETURN

```

```

215 IF A = 8 AND LEN (I$) > 0 THEN I$ = MID$ (I$,1,LEN (I$) - 1):PRINT A$;CHR$ (95);A$;
220 IF A = 8 THEN 205
225 IF LEN (I$) = MX THEN 205
230 IF A < 32 OR A > 90 THEN 205
235 PRINT A$;
240 I$ = I$ + A$: GOTO 205
250 MM = 0:DD = 0:YY = 0
255 K = LEN (I$):IF K < 6 OR K > 8 THEN 305
260 MM = VAL (I$):FOR J = 1 TO K
265 IF VAL (MID$ (I$,J,1)) = 0 AND MID$ (I$,J,1) <
> "0" THEN DD = YY:YY = J + 1
270 NEXT :IF DD = 0 THEN 305
275 DD = VAL (MID$ (I$,DD,2)):YY = VAL (MID$ (I$,YY)
)
280 IF MM < 1 OR MM > 12 THEN 305
285 IF DD < 1 OR DD > 31 THEN 305
290 IF MM = 2 AND (INT (YY / 4) * 4 < > YY) AND (INT (YY / 400) * 400 < > YY) AND DD > 28 THEN 305
295 IF (MM = 4 OR MM = 6 OR MM = 9 OR MM = 11) AND DD > 30 THEN 305
300 I$ = RIGHT$ ("00" + STR$ (MM),2) + "/" + RIGHT$ ("00" + STR$ (DD),2) + "/" + RIGHT$ ("00" + STR$ (YY),2):RETURN
305 I$ = "":RETURN
325 GET CH$:IF CH$ < > "Y" AND CH$ < > "N" THEN 325
PRINT CH$:RETURN
330 FOR X = 1 TO MX:PRINT CHR$ (95);:NEXT :FOR X = 1 TO MX:PRINT CHR$ (8);:NEXT :RETURN
400 PRINT :PRINT "SORRY, INVALID RECORD NUMBER 1":FOR X = 1 TO 1000:NEXT :RETURN
500 HOME
510 PRINT "TODAY'S DATE (MM/DD/YY): ";:MX = 8:GOSUB 350:GOSUB 200:GOSUB 250:IF I$ = "" THEN 510
520 DT$ = I$
600 TEXT :HOME
610 PRINT TAB( 9)"MASTER PROGRAM CATALOG":PRINT TAB( 16)DT$
620 POKE 32,8
630 PRINT :PRINT
640 PRINT "1) ENTER LISTING"
650 PRINT "2) CHANGE LISTING"
660 PRINT "3) DELETE LISTING"
670 PRINT "4) SEARCH LISTINGS"
680 PRINT "5) PRINT LISTINGS"
690 PRINT "6) SORT LISTINGS"
700 PRINT "7) EXIT PROGRAM"
710 POKE 32,0
720 VTAB 20:PRINT "CHOICE (1-7): ";
730 GET CH$:CH = VAL (CH$):IF CH < 1 OR CH > 7 THEN 730
740 PRINT :ON CH GOTO 1000,2000,3000,4000,5000,6000,7000
1000 HOME

```

```

1010 PRINT "THIS WILL BE PROGRAM # ";N + 1:PRINT
1020 PRINT "WHAT IS THE PROGRAM NAME ?"
1030 MX = 35:GOSUB 350:GOSUB 200:T$(N + 1,1) = I$
1040 VTAB 7:PRINT "LANGUAGE TYPE: ";
1050 MX = 10:GOSUB 350:GOSUB 200:T$(N + 1,2) = I$
1060 VTAB 9:PRINT "PROGRAM TYPE:":PRINT
1070 FOR X = 1 TO NT:PRINT " ";X;" ";PT$(X):NEX
T :PRINT
1080 MX = 1:GOSUB 350:GOSUB 200:T = VAL (I$):IF T < 1
OR T > NT THEN 1060
1090 T$(N + 1,3) = I$
1100 T$(N + 1,4) = DT$
1110 PRINT :PRINT "DISK LOCATION: ";
1120 MX = 5:GOSUB 350:GOSUB 200:T$(N + 1,5) = I$
1130 N = N + 1
1140 PRINT :PRINT "MORE ENTRIES (Y/N): ";
1150 GOSUB 325:IF CH$ = "Y" THEN 1000
1160 GOTO 600
2000 IF N = 0 THEN PRINT "SORRY, THERE ARE NO RECORDS
...":FOR J = 1 TO 1000:NEXT : GOTO 600
2010 HOME :PRINT "WHAT IS THE RECORD NUMBER OF THE LI
STINGYOU WISH TO CHANGE ? "
2020 MX = 4:GOSUB 350
2030 GOSUB 200:IF I$ = "" THEN 600
2040 CH = VAL (I$):IF CH < 1 OR CH > N THEN GOSUB 400
: GOTO 2000
2050 HOME
2060 PRINT "RECORD NUMBER: ";CH
2070 PRINT :FOR X = 1 TO 5:PRINT X;".":PRINT :NEXT :P
OKE 32,4
2080 VTAB 3:PRINT "NAME: ";T$(CH,1)
2090 PRINT :PRINT "LANGUAGE: ";T$(CH,2)
2100 PRINT :PRINT "PGM TYPE: ";PT$(VAL (T$(CH,3)))
2110 PRINT :PRINT "DATE ENTERED: ";T$(CH,4)
2120 PRINT :PRINT "DISK LOCATION: ";T$(CH,5)
2130 POKE 32,0:PRINT :PRINT
2140 PRINT :PRINT "NUMBER OF ITEM TO CHANGE (1-5): ";
2150 GET CH$:Q = VAL (CH$):IF Q < 1 OR Q > 5 THEN 215
0
2160 PRINT Q:PRINT :PRINT "CHANGE TO WHAT: ";
2170 IF Q = 1 THEN MX = 35:PRINT
2180 IF Q = 2 THEN MX = 10
2190 IF Q = 3 THEN MX = 1
2200 IF Q = 4 THEN MX = 8
2210 IF Q = 5 THEN MX = 5
2220 GOSUB 350:GOSUB 200
2230 IF Q = 4 THEN GOSUB 250
2240 T$(CH,Q) = I$
2250 PRINT :PRINT "MORE CHANGES (Y/N): ";:GOSUB 325:I
F CH$ = "Y" THEN 2000
2260 GOTO 600
3000 IF N = 0 THEN PRINT "SORRY, THERE ARE NO RECORDS
...":FOR J = 1 TO 1000:NEXT : GOTO 600
3010 HOME :PRINT "WHAT IS THE RECORD NUMBER OF THE LI
STINGYOU WISH TO DELETE ? "

```

```

3020 MX = 4:GOSUB 350
3030 GOSUB 200
3040 CH = VAL (I$):IF CH < 1 OR CH > N THEN GOSUB 400
      : GOTO 3000
3050 HOME
3060 PRINT "RECORD NUMBER: ";CH
3070 PRINT :FOR X = 1 TO 5:PRINT X;".":PRINT :NEXT :P
      OKE 32,4
3080 VTAB 3:PRINT "NAME: ";T$(CH,1)
3090 PRINT :PRINT "LANGUAGE: ";T$(CH,2)
3100 PRINT :PRINT "PGM TYPE: ";PT$(VAL (T$(CH,3)))
3110 PRINT :PRINT "DATE ENTERED: ";T$(CH,4)
3120 PRINT :PRINT "DISK LOCATION: ";T$(CH,5)
3130 POKE 32,0:PRINT :PRINT
3140 PRINT "O.K. TO DELETE (Y/N): ";:GOSUB 325:IF CH$
      = "N" THEN 600
3150 FOR X = 1 TO 5:T$(CH,X) = T$(N,X):NEXT :N = N -
      1: GOTO 600
4000 IF N = 0 THEN PRINT "SORRY, THERE ARE NO RECORDS
      ...":FOR J = 1 TO 1000:NEXT : GOTO 600
4010 HOME:PRINT "WOULD YOU LIKE TO SEARCH BY:":PRINT
4020 PRINT "      1) RECORD NUMBER"
4030 PRINT "      2) PROGRAM NAME"
4040 PRINT "      3) LANGUAGE"
4050 PRINT "      4) PROGRAM TYPE"
4060 PRINT "      5) DATE ENTERED"
4070 PRINT "      6) DISK LOCATION"
4080 PRINT :PRINT :PRINT "WHICH CHOICE (1-6): ";
4090 GET CH$:CH = VAL (CH$):IF CH < 1 OR CH > 6 THEN
      4090
4100 PRINT CH:PRINT :PRINT "KEYWORD FOR SEARCH: ";
4110 IF CH = 1 THEN MX = 4
4120 IF CH = 2 THEN MX = 35:PRINT
4130 IF CH = 3 THEN MX = 10
4140 IF CH = 4 THEN MX = 1
4150 IF CH = 5 THEN MX = 8
4160 IF CH = 6 THEN MX = 5
4170 GOSUB 350:GOSUB 200:Q$ = I$
4180 IF CH = 5 THEN GOSUB 250:Q$ = I$:IF Q$ = "" THEN
      PRINT :PRINT "ILLEGAL DATE FORMAT!":FOR X = 1
      TO 1000:NEXT : GOTO 4000
4190 HOME :PRINT TAB( 15):FLASH :PRINT "SEARCHING":NO
      RMAL :POKE 34,2
4200 IF CH = 1 THEN 4380
4210 FOR X = 1 TO N
4220 IF T$(X,CH - 1) < > Q$ THEN 4330
4230 HOME
4240 PRINT :PRINT "RECORD NUMBER: ";X
4250 PRINT :PRINT "NAME: ";T$(X,1)
4260 PRINT :PRINT "LANGUAGE: ";T$(X,2)
4270 PRINT :PRINT "PGM TYPE: ";PT$(VAL (T$(X,3)))
4280 PRINT :PRINT "DATE ENTERED: ";T$(X,4)
4290 PRINT :PRINT "DISK LOCATION: ";T$(X,5)
4300 IF CH = 1 THEN 4340
4310 VTAB 23:PRINT "PRESS <RETURN> TO CONTINUE...";

```

```

4320 GET P$:IF P$ < > CHR$(13) THEN 4320
4330 NEXT
4340 POKE 34,0
4350 VTAB 23:HTAB 1:CALL - 958:FLASH :PRINT "END OF
SEARCH...";:NORMAL :PRINT "PRESS <RETURN>...";
4360 GET P$:IF P$ < > CHR$(13) THEN 4360
4370 GOTO 600
4380 X = VAL (Q$):IF X < 1 OR X > N THEN VTAB 10:GOSU
B 400: GOTO 600
4390 PRINT :PRINT "RECORD NUMBER: ";X
4400 GOTO 4230
5000 IF N = 0 THEN PRINT "SORRY, THERE ARE NO RECORDS
...":FOR J = 1 TO 1000:NEXT : GOTO 600
5010 C = 0:FOR X = 1 TO N
5020 C = C + 1:IF C = 1 THEN HOME
5030 INVERSE :PRINT X;:NORMAL :PRINT " ";T$(X,1)
5040 PRINT " ";T$(X,2);" ";PT$(VAL (T$(X,3)));" "
;T$(X,4);" ";T$(X,5)
5050 PRINT
5060 IF C < 7 THEN 5100
5070 VTAB 23:PRINT "PRESS <RETURN> TO CONTINUE...";
5080 GET CH$:IF CH$ < > CHR$(13) THEN 5080
5090 VTAB 23:HTAB 1:PRINT C = 0
5100 NEXT
5110 VTAB 23:HTAB 1:CALL - 958:PRINT "END OF LISTING
...PRESS <RETURN>...";
5120 GET CH$:IF CH$ < > CHR$(13) THEN 5120
5130 GOTO 600
6000 IF N = 0 THEN PRINT "SORRY, THERE ARE NO RECORDS
...":FOR J = 1 TO 1000:NEXT : GOTO 600
6005 HOME :PRINT "WOULD YOU LIKE TO SORT BY:":PRINT
6010 PRINT " 1) PROGRAM NAME"
6020 PRINT " 2) LANGUAGE"
6030 PRINT " 3) PROGRAM TYPE"
6040 PRINT " 4) DATE ENTERED"
6050 PRINT " 5) DISK LOCATION"
6060 PRINT :PRINT :PRINT "WHICH CHOICE (1-5): ";
6070 GET CH$:CH = VAL (CH$):IF CH < 1 OR CH > 5 THEN
6070
6080 PRINT CH:HOME :VTAB 8:HTAB 10:PRINT "SORTING - O
NE MOMENT!"
6090 REM - QUICKSORT
6100 P = 1:Q = N:T0 = 0
6110 IF P > = Q THEN 6200
6120 V$ = T$(P,CH):I = P:J = Q + 1
6130 J = J - 1:IF T$(J,CH) > V$ THEN 6130
6140 I = I + 1:IF T$(I,CH) < V$ AND I < N THEN 6140
6150 IF J > I THEN FOR X = 1 TO 5:T$ = T$(I,X):T$(I,X
) = T$(J,X):T$(J,X) = T$:NEXT : GOTO 6130
6160 FOR X = 1 TO 5:T$ = T$(P,X):T$(P,X) = T$(J,X):T$
(J,X) = T$:NEXT
6170 IF (J - P) < (Q - J) THEN ST(T0 + 1) = J + 1:ST(
T0 + 2) = Q:Q = J - 1: GOTO 6190
6180 ST(T0 + 1) = P:ST(T0 + 2) = J - 1:P = J + 1

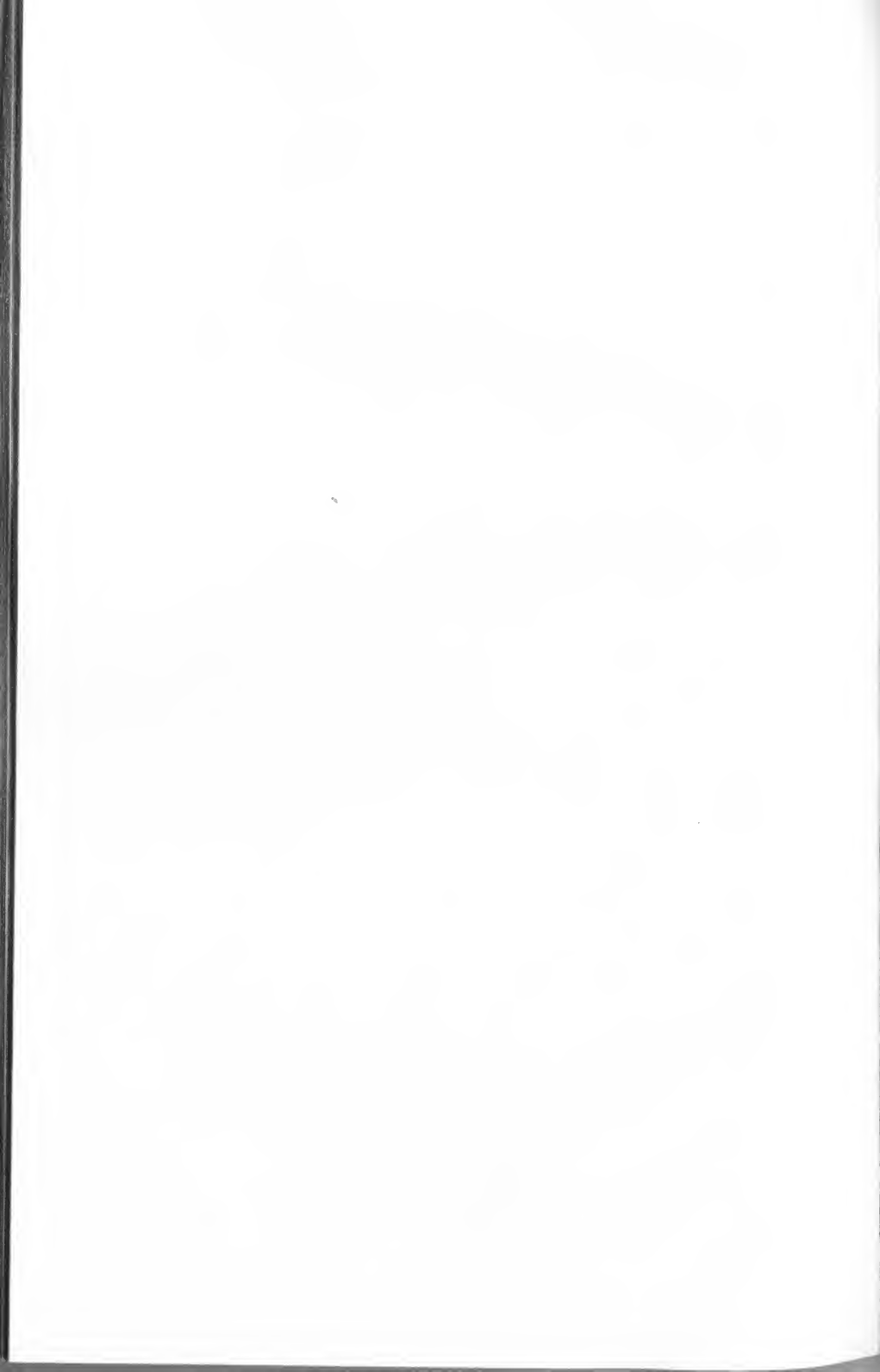
```

```

6190 T0 = T0 + 2: GOTO 6110
6200 IF T0 < > 0 THEN Q = ST(T0):P = ST(T0 - 1):T0 =
      T0 - 2: GOTO 6110
6210 GOTO 600
7000 PRINT OP$;"PROGRAMS":PRINT D$;"DELETE PROGRAMS":
      PRINT OP$;"PROGRAMS"
7010 PRINT WR$;"PROGRAMS"
7020 PRINT N:IF N = 0 THEN 7070
7030 FOR X = 1 TO N
7040 FOR Y = 1 TO 5
7050 PRINT T$(X,Y)
7060 NEXT :NEXT
7070 PRINT CL$
7080 PRINT "LOCK PROGRAMS"
7090 HOME :PRINT :PRINT D$;"RUN MENU"

```







More Books for Apple Owners!

INTRODUCING THE APPLE® MACINTOSH™

Introduces you to the design philosophy and physical structure of the Macintosh™, and explores its displays, keyboard, mouse, software, accessories, and more. By Connolly and Lieberman. 192 pages, 8 x 9 1/4, softbound. ISBN 0-672-22361-9. © 1984.

No. 22361\$12.95

APPLE® IIe PROGRAMMERS' REFERENCE GUIDE

An outstanding IIe reference guide specifically for the IIe that makes needed facts, applications, and other technical information readily available at your fingertips. By David L. Heiserman. 416 pages, 5 1/2 x 8 1/2, comb-bound. ISBN 0-672-22299-X. © 1984.

No. 22299\$19.95

APPLESOFT FOR THE IIe

A detailed Applesoft programmer's reference manual written specifically for the IIe and covering all aspects of IIe syntax and programming techniques. By Blackwood and Blackwood. 368 pages, 6 x 9, comb-bound. ISBN 0-672-22259-0. © 1983.

No. 22259\$19.95

APPLE® PROGRAMMER'S HANDBOOK

This single-volume coverage of essential Apple data contains dozens of tested "stock" routines organized by topic, a detailed memory map, and much more. By Paul Irwin. 480 pages, 5 1/2 x 8 1/2, comb-bound. ISBN 0-672-22175-6. © 1984.

No. 22175\$21.95

BASIC TRICKS FOR THE APPLE®

From a seasoned professional comes this collection of ideas, examples, and special Applesoft subroutines to use or modify as part of your own Apple programs. By Allen L. Wyatt. 160 pages, 5 1/2 x 8 1/2, softbound. ISBN 0-672-22208-6. © 1983.

No. 22208\$8.95

APPLE® II FOR KIDS FROM 8 TO 80

Whatever your age, you'll think you're at Computer Camp as these enjoyable and easy to follow, beginner-level BASIC programming instructions help you quickly begin writing your own Apple II-compatible programs. By Michael P. Zabinski and Frank Mazzola. 160 pages, 8 1/2 x 11, softbound. ISBN 0-672-22297-3. © 1984.

No. 22297\$10.95

ENHANCING YOUR APPLE® II, Volume 1 (2nd Edition)

Lets you mix text, low-res, and high-res together anywhere on-screen; have 3-D graphics, overlapping single-line colors, and other special effects; tear apart and understand somebody else's machine-language program, and much more. By Don Lancaster. 256 pages, 8 1/2 x 11, softbound. ISBN 0-672-21822-4. © 1984.

No. 21822\$15.95

THE APPLE® II CIRCUIT DESCRIPTION

Gives you a detailed circuit description of all revisions of the Apple II and Apple II+ motherboard, including the keyboard and power supply. By Winston D. Gayler. 176 pages plus foldouts, 8½ x 11, comb-bound. ISBN 0-672-21959-X. © 1983.

No. 21959\$22.95

DISKS, FILES, AND PRINTERS FOR THE APPLE® II

Provides basic-to-advanced details for using disks, files, and printers with an Apple II, plus hard-to-find advice on programming with sequential-access, random-access, and executive files. By Blackwood and Blackwood. 216 pages, 6 x 9, comb-bound. ISBN 0-672-22163-2. © 1983.

No. 22163\$15.95

APPLE® II APPLICATIONS

Gives you a broad spectrum of tested programming and board-level interfacing applications, including serial and parallel I/O boards, EPROM or E²PROM boards, remote data acquisition, and more. By Marvin L. De Jong. 256 pages, 5½ x 8½, softbound. ISBN 0-672-22035-0. © 1983.

No. 22035\$13.95

APPLESOFT LANGUAGE (2nd Edition)

New material quickly introduces you to Applesoft syntax and programming, including advanced programming techniques, graphics, color commands, sorts, searches, and more. By Blackwood and Blackwood. 288 pages, 6 x 9, comb-bound. ISBN 0-672-22073-3. © 1983.

No. 22073\$13.95

MOSTLY BASIC: APPLICATIONS FOR YOUR APPLE® II, Book 1

Twenty-eight Applesoft programs, including a telephone dialer, digital stopwatch, a spelling test, house-buying guide, gas mileage calculator, and many more. By Howard Berenbon. 160 pages, 8½ x 11, comb-bound. ISBN 0-672-21789-9. © 1980.

No. 21789\$13.95

MOSTLY BASIC: APPLICATIONS FOR YOUR APPLE® II, Book 2

More fascinating BASIC programs, including three dungeons, eleven household programs, seven on money or investment, two that test your level of ESP, and more — 32 in all! By Howard Berenbon. 224 pages, 8½ x 11, comb-bound. ISBN 0-672-21864-X. © 1981.

Ask for No. 21864\$12.95

INTERMEDIATE LEVEL APPLE® II HANDBOOK

Hard-to-find, practical info that uses ROM-based Integer BASIC to lead you into Apple 6502 machine and assembly language programming. By David L. Heiserman. 328 pages, 6 x 9, comb-bound. ISBN 0-672-21889-5. © 1983.

No. 21889\$16.95

INTIMATE INSTRUCTIONS IN INTEGER BASIC

Includes much to help you build Integer programs that run smoothly and take full advantage of that dialect's rapid-running characteristics. By Blackwood and Blackwood. 160 pages, 5½ x 8½, soft. ISBN 0-672-21812-7. © 1981.

No. 21812\$8.95

These and other Sams Books and Software products are available from better retailers worldwide, or directly from Sams. Call 800-428-SAMS or 317-298-5566 to order, or to get the name of a Sams retailer near you. Ask for your free Sams Books and Software catalog!

Prices good in USA only. Prices and page counts subject to change without notice.

Apple is a registered trademark of Apple Computer, Inc. Macintosh is a trademark of Apple Computer, Inc.



Apple® Games

- Gives Apple users 11 fun-packed games
- Includes 3 other miscellaneous programs
- Provides helpful loading instructions
- Teaches how to use text, graphics, and sounds in developing your own computer games
- Complete program listings provided for every program

Apple® Games is not only fun, it's educational. See for yourself how these programs work and try your hand at adapting them to produce different special effects.

Machine Requirements:

Apple® II Compatible Systems (48K)

Applesoft in ROM

1 Disk Drive (DOS 3.3)

Game Paddles or Joysticks required on selected games

Howard W. Sams & Co., Inc.

4300 West 62nd Street, Indianapolis, Indiana 46268 U.S.A.

\$8.95/22394

ISBN: 0-672-22394-7