

25 DAZZLING PROGRAM DISPLAYS

ART & GRAPHICS

WITH YOUR
TI-99/4A™



Thomas A. Thompson, Jr.

HAYDEN

ART & GRAPHICS **WITH YOUR** **TI-99/4A™**

Thomas A. Thompson, Jr.



HAYDEN BOOK COMPANY
a division of Hayden Publishing Company, Inc.
Hasbrouck Heights, New Jersey

EQUIPMENT NEEDED

To use the programs in the book on a Texas Instruments computer, you will need the following equipment:

- A TI-99/4A Home Computer
- A television or monitor (preferably color)
- A data cassette recorder (optional)

Acquisitions Editor: JONO HARDJO

Production Editor: MAUREEN CONNELLY

Text Design: SHARYN BANKS

Cover Design: JIM BERNARD

Cover Sculpture: SHARYN BANKS

Compositor: AMERICAN-STRATFORD GRAPHIC SERVICES, INC.

Printed and bound by: THE BOOK PRESS

Library of Congress Cataloging in Publication Data

Thompson, Thomas A.

Art and graphics with your TI-99/4A.

1. Computer graphics. 2. TI 99/4A (Computer)—Programming. I. Title.

T385.T4954 1984 001.64'43 84-3760
ISBN 0-8104-6407-1

TI-99/4A is a trademark of Texas Instruments Incorporated, which is not affiliated with Hayden Book Company.

Copyright © 1984 by Hayden Book Company. All rights reserved. No part of this book may be reprinted, or reproduced, or utilized in any form or by an electronic, mechanical, or other means, now known or hereafter invented, including photocopying and recording, or in any information storage and retrieval system, without permission in writing from the Publisher.

Printed in the United States of America

1	2	3	4	5	6	7	8	9	PRINTING
84	85	86	87	88	89	90	91	92	YEAR

INTRODUCTION

This book is intended for all those who purchase a Texas Instruments 99/4A and wish to do more with it than just plug in modules. The programs on the following pages can show you the way to open the secret door contained in your computer. You already have the key to that door: it's your mind. Behind that door is a world of sights and sounds never before experienced.

By running the enclosed programs, you will learn how to put life into your computer. The computer needs only a few simple instructions. There are several ways to create movement used in this book. They include: changing coordinates of a character on the screen, CALL HCHAR, CALL VCHAR, CALL KEY, CALL COLOR, and PRINT. Many of the programs are designed to run continuously. These may be disrupted by pressing FCTN-4.

The games and graphics in this book are done in such a way as to be fun to do and watch as they are, yet they leave room for alterations. You may add or subtract using your imagination. So sit back, start typing, and open wide that secret door.

CONTENTS

Ol' Glory	1
Here Comes Santa	3
Valentine Message	7
For Sail	10
Oops, Don't Look	13
Tunnel Vision	16
Blind as a Bat	19
Danger in the Deep	22
I Saw the Light	25
Up, Up, and Away	28
Spy in the Sky	31
Out of this World	35
Just a Little Squirt	39
There She Blows	43
Fly Trap	46
Just Good Ol' Boys	52
Play Ball	54
Fresh Eggs Today	59
Bird Watching	62
Face to Face	65
Snake Eyes	71
Just One Bad Apple	74
Skulduggery	79
U Turn	83
Hang 'Em High	86
Appendix A—Command Glossary	93
Appendix B—Hexidecimal Code	95
Appendix C—ASCII Character Code	96
Appendix D—Color Codes	98

OL' GLORY

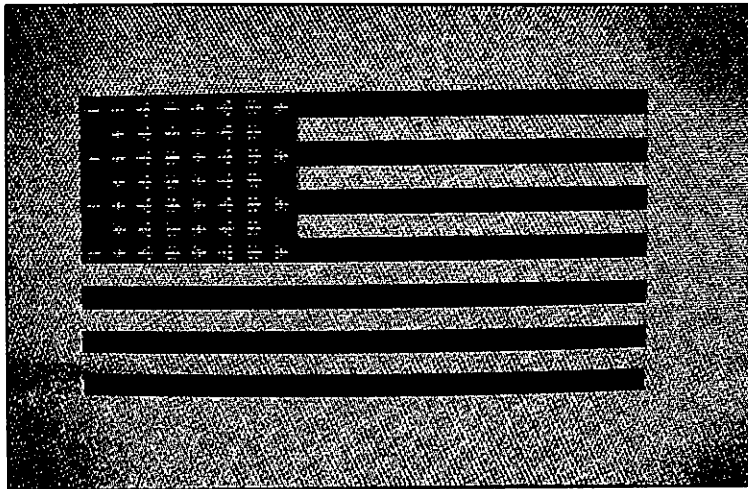
Let's start off our graphics with an easy nonmoving art work. OL' GLORY uses CALL HCHAR exclusively in order to produce a familiar American symbol.

PROGRAM STRUCTURE

6-12	set the colors
14-16	define characters
18-70	place characters on screen
72	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Try changing the number in the CALL SCREEN statement. Watch how different screen colors change the hue of colors on the flag.




```
2 REM OL' GLORY
4 CALL CLEAR
6 CALL SCREEN(12)
8 CALL COLOR(2,15,5)
10 CALL COLOR(3,9,9)
12 CALL COLOR(4,15,15)
14 CALL CHAR(48,"FFFFFFFFFFFFFFFF")
16 CALL CHAR(56,"FFFFFFFFFFFFFFFF")
18 CALL CHAR(43,"00")
20 CALL HCHAR(5,6,42,8)
22 CALL HCHAR(5,14,48,13)
24 CALL HCHAR(6,6,43)
26 CALL HCHAR(6,7,42,6)
28 CALL HCHAR(6,13,43)
30 CALL HCHAR(6,14,56,13)
32 CALL HCHAR(7,6,42,8)
34 CALL HCHAR(7,14,48,13)
36 CALL HCHAR(8,6,43)
38 CALL HCHAR(8,7,42,6)
40 CALL HCHAR(8,13,43)
42 CALL HCHAR(8,14,56,13)
44 CALL HCHAR(9,6,42,8)
46 CALL HCHAR(9,14,48,13)
48 CALL HCHAR(10,6,43)
50 CALL HCHAR(10,7,42,6)
52 CALL HCHAR(10,13,43)
54 CALL HCHAR(10,14,56,13)
56 CALL HCHAR(11,6,42,8)
58 CALL HCHAR(11,14,48,13)
60 CALL HCHAR(12,6,56,21)
62 CALL HCHAR(13,6,48,21)
64 CALL HCHAR(14,6,56,21)
66 CALL HCHAR(15,6,48,21)
68 CALL HCHAR(16,6,56,21)
70 CALL HCHAR(17,6,48,21)
72 GOTO 12
74 END
```

HERE COMES SANTA

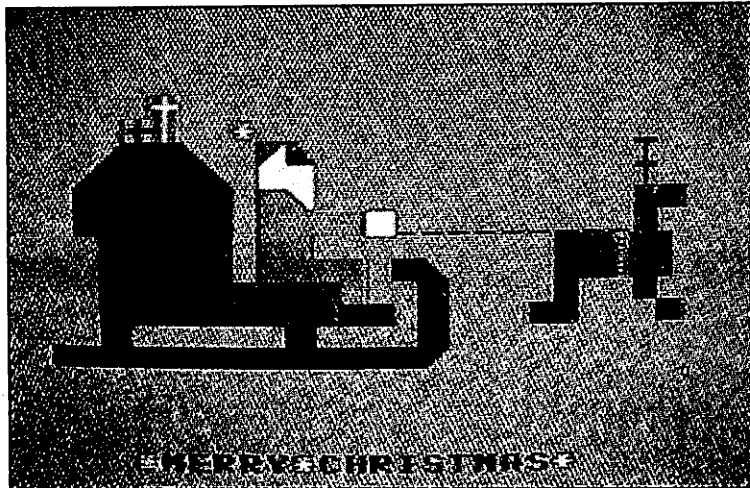
Santa's coming and gifts are not the only things he's bringing—this time he also carries a tune!

PROGRAM STRUCTURE

15-45	set colors
50-130	define characters
135-350	place Santa on screen
355-420	display words on screen
425-430	time delay
435-490	control snowflakes and music
495	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Add your own favorite Christmas carol, or include a more personal Christmas greeting.



```

1 REM   HERE COMES SANTA
5 CALL CLEAR
10 CALL SCREEN(6)
15 CALL COLOR(2,16,1)
20 CALL COLOR(8,2,1)
25 CALL COLOR(9,12,13)
30 CALL COLOR(10,9,16)
35 CALL COLOR(13,2,1)
40 CALL COLOR(14,7,1)
45 CALL COLOR(15,9,16)
50 CALL CHAR(97,"E7E7E70000E7E7E7")
55 CALL CHAR(104,"E7E7E7E7E7E7E7E7")
60 CALL CHAR(105,"E7E7E70000E7E7E7")
65 CALL CHAR(128,"FFFFFFFFFFFFFFFF")
70 CALL CHAR(131,"0103070F1F3F7FFF")
75 CALL CHAR(129,"80C0E0F0F8FCFEFF")
80 CALL CHAR(130,"FFFEFCF8F0E0C080")
85 CALL CHAR(136,"FFFFFFFFFFFFFFFF")
90 CALL CHAR(137,"FFFCFCFFFFFFFFFFFF")
95 CALL CHAR(138,"E3E3EBE3EBE3EBE3")
100 CALL CHAR(139,"F0F0F0F0F0F0F0F0")
105 CALL CHAR(140,"FFFF1B1B1B1B1B1B")
110 CALL CHAR(142,"E0F0C8FCFEFEFFFF")
115 CALL CHAR(144,"FFFFFFFFFFFFFFFF")
120 CALL CHAR(145,"0000000000000000")
125 CALL CHAR(146,"80C0E0F0F8FCFEFF")
130 CALL CHAR(147,"FFFEFCF8F0E0C080")
135 CALL HCHAR(8,7,128,2)
140 CALL HCHAR(8,9,129)
145 CALL HCHAR(9,10,129)
150 CALL HCHAR(9,6,128,4)
155 CALL HCHAR(8,6,131)
160 CALL HCHAR(9,5,131)
165 CALL HCHAR(10,5,128,6)
170 CALL HCHAR(11,5,128,6)
175 CALL HCHAR(17,4,128,14)
180 CALL VCHAR(14,18,128,3)
185 CALL VCHAR(13,17,128)
190 CALL HCHAR(7,7,97)
195 CALL HCHAR(7,8,104)
200 CALL HCHAR(6,8,105)
205 CALL VCHAR(13,18,129)
210 CALL HCHAR(17,18,130)
215 CALL HCHAR(15,15,128,2)
220 CALL HCHAR(12,6,136,5)
225 CALL HCHAR(13,6,136,5)
230 CALL HCHAR(14,6,136,9)
235 CALL HCHAR(15,6,136,9)

```

```
240 CALL HCHAR(16,6,128)
245 CALL HCHAR(16,13,128)
250 CALL HCHAR(15,22,136,2)
255 CALL HCHAR(15,27,136)
260 CALL HCHAR(10,27,136)
265 CALL VCHAR(11,26,136,4)
270 CALL VCHAR(12,23,136,3)
275 CALL VCHAR(12,24,136,2)
280 CALL VCHAR(10,26,137)
285 CALL VCHAR(12,25,138,2)
290 CALL VCHAR(12,27,139,2)
295 CALL VCHAR(8,26,140,2)
300 CALL VCHAR(10,12,144,3)
305 CALL VCHAR(11,13,144,3)
310 CALL HCHAR(13,12,144,4)
315 CALL HCHAR(14,15,144)
320 CALL HCHAR(11,14,144,2)
325 CALL HCHAR(9,12,145,2)
330 CALL HCHAR(11,16,145)
335 CALL VCHAR(10,13,146)
340 CALL VCHAR(8,12,147)
345 CALL VCHAR(7,11,42)
350 CALL VCHAR(8,13,142)
355 CALL HCHAR(11,17,95,9)
360 CALL HCHAR(22,8,77)
365 CALL HCHAR(22,9,69)
370 CALL HCHAR(22,10,82,2)
375 CALL HCHAR(22,12,89)
380 CALL HCHAR(22,14,67)
385 CALL HCHAR(22,15,72)
390 CALL HCHAR(22,16,82)
395 CALL HCHAR(22,17,73)
400 CALL HCHAR(22,18,83)
405 CALL HCHAR(22,19,84)
410 CALL HCHAR(22,20,77)
415 CALL HCHAR(22,21,65)
420 CALL HCHAR(22,22,83)
425 FOR DELAY=1 TO 700
430 NEXT DELAY
435 CALL HCHAR(22,7,42,17)
440 CALL SOUND(400,659,3)
445 CALL SOUND(400,659,3)
450 CALL SOUND(800,659,2)
455 CALL SOUND(400,659,3)
460 CALL SOUND(400,659,3)
465 CALL SOUND(800,659,2)
```

```
470 CALL SOUND(400,659,3)
475 CALL SOUND(400,784,2)
480 CALL SOUND(500,523,3)
485 CALL SOUND(300,587,4)
490 CALL SOUND(700,659,3)
495 GOTO 355
500 END
```

VALENTINE MESSAGE

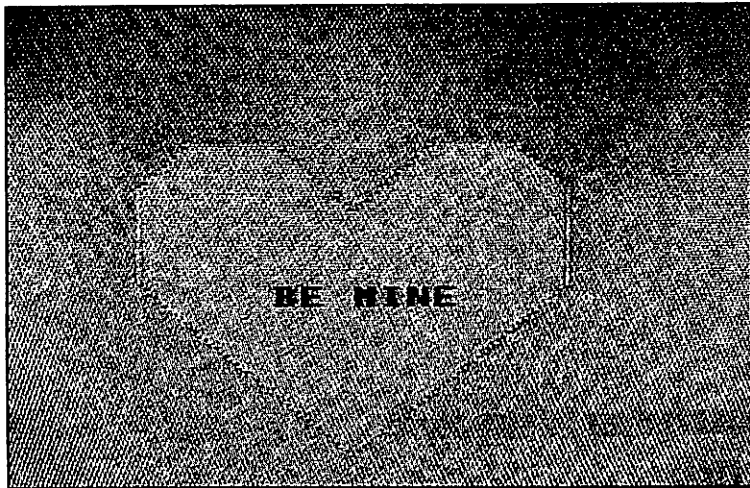
Here is a new way to tell that special someone something special.

PROGRAM STRUCTURE

15-35	define characters
40-70	set colors
75-290	place heart on screen
335-470	control messages

RECOMMENDATIONS FOR ALTERATIONS

Put your own personal messages into the program. Change the background color to suit your mood.




```

5 REM    VALENTINE MESSAGE
10 CALL CLEAR
15 CALL CHAR(40,"FFFFFFFFFFFFFFFF")
20 CALL CHAR(41,"0103070F1F3F7FFF")
25 CALL CHAR(42,"80E0C0F0F8FCFEFF")
30 CALL CHAR(43,"FF7F3F1F0F070301")
35 CALL CHAR(44,"FFFEFCF8F0E0C080")
40 CALL COLOR(2,9,1)
45 CALL COLOR(5,2,9)
50 CALL COLOR(6,2,9)
55 CALL COLOR(7,2,9)
60 CALL COLOR(8,2,9)
65 CALL COLOR(9,2,9)
70 CALL SCREEN(6)
75 CALL VCHAR(10,8,40,4)
80 CALL VCHAR(9,8,41)
85 CALL VCHAR(9,9,40,6)
90 CALL VCHAR(8,9,41)
95 CALL VCHAR(8,10,40,8)
100 CALL VCHAR(14,8,43)
105 CALL VCHAR(8,11,40,9)
110 CALL VCHAR(8,12,40,10)
115 CALL VCHAR(8,13,42)
120 CALL VCHAR(15,9,43)
125 CALL VCHAR(9,13,40,10)
130 CALL VCHAR(9,14,42)
135 CALL VCHAR(10,14,40,10)
140 CALL VCHAR(10,15,42)
145 CALL VCHAR(16,10,43)
150 CALL VCHAR(11,15,40,10)
155 CALL VCHAR(11,16,40,10)
160 CALL VCHAR(10,16,41)
165 CALL VCHAR(17,11,43)
170 CALL VCHAR(11,16,40,10)
175 CALL VCHAR(21,16,44)
180 CALL VCHAR(10,17,40,10)
185 CALL VCHAR(9,17,41)
190 CALL VCHAR(20,17,44)
195 CALL VCHAR(18,12,43)
200 CALL VCHAR(8,18,41)
205 CALL VCHAR(9,18,40,10)
210 CALL VCHAR(19,18,44)
215 CALL VCHAR(8,19,40,10)
220 CALL VCHAR(18,19,44)
225 CALL VCHAR(19,13,43)
230 CALL VCHAR(8,20,40,9)
235 CALL VCHAR(8,21,40,8)
240 CALL VCHAR(20,14,43)
245 CALL VCHAR(8,22,42)

```

```
250 CALL VCHAR(9,22,40,6)
255 CALL VCHAR(9,23,42)
260 CALL VCHAR(10,23,40,4)
265 CALL VCHAR(21,15,43)
270 CALL VCHAR(17,20,44)
275 CALL VCHAR(16,21,44)
280 CALL VCHAR(15,22,44)
285 CALL VCHAR(14,23,44)
290 CALL HCHAR(19,17,40)
295 GOSUB 335
300 GOSUB 355
305 GOSUB 335
310 GOSUB 390
315 GOSUB 335
320 GOSUB 430
325 GOTO 295
330 STOP
335 FOR DELAY=1 TO 600
340 NEXT DELAY
345 CALL HCHAR(14,12,40,9)
350 RETURN
355 CALL HCHAR(14,13,66)
360 CALL HCHAR(14,14,69)
365 CALL HCHAR(14,16,77)
370 CALL HCHAR(14,17,73)
375 CALL HCHAR(14,18,78)
380 CALL HCHAR(14,19,69)
385 RETURN
390 CALL HCHAR(14,12,76)
395 CALL HCHAR(14,13,79)
400 CALL HCHAR(14,14,86)
405 CALL HCHAR(14,15,69)
410 CALL HCHAR(14,18,89)
415 CALL HCHAR(14,19,79)
420 CALL HCHAR(14,20,85)
425 RETURN
430 CALL HCHAR(14,12,73)
435 CALL HCHAR(14,13,96)
440 CALL HCHAR(14,14,77)
445 CALL HCHAR(14,16,89)
450 CALL HCHAR(14,17,79)
455 CALL HCHAR(14,18,85)
460 CALL HCHAR(14,19,82)
465 CALL HCHAR(14,20,83)
470 RETURN
475 END
```

FOR SAIL

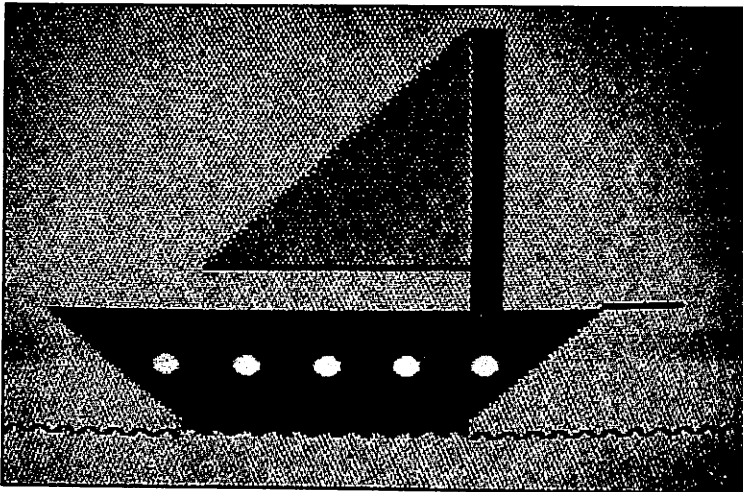
If gentle rolling waves upset your stomach, then pass this program up. If not, batten down the hatches and get set to heave to!

PROGRAM STRUCTURE

6-16	set colors
18-42	define characters
44-126	place the boat on the screen
128-132, 138-142	control the waves
134-136, 144-146	time delays
148	keeps the program running

RECOMMENDATIONS FOR ALTERATIONS

Have a sea gull in the sky or a fish in the water.



```

2 REM   FOR SAIL
4 CALL CLEAR
6 CALL SCREEN(6)
8 CALL COLOR(9,2,1)
10 CALL COLOR(10,9,1)
12 CALL COLOR(11,13,1)
14 CALL COLOR(12,15,13)
16 CALL COLOR(13,6,13)
18 CALL CHAR(97,"0814224180000000")
20 CALL CHAR(98,"FFFFFFFFFFFFFFFF")
22 CALL CHAR(99,"FFFF")
24 CALL CHAR(100,"000000000000FFFF")
26 CALL CHAR(101,"0000000000384483")
28 CALL CHAR(104,"0103070F1F3F7FFF")
30 CALL CHAR(105,"FFFFFFFFFFFFFFFF")
32 CALL CHAR(112,"FF7F3F1F0F070301")
34 CALL CHAR(113,"FFFEFCF8F0E0C080")
36 CALL CHAR(114,"FFFFFFFFFFFFFFFF")
38 CALL CHAR(120,"3C7EFFFFFFFF7E3C")
40 CALL CHAR(128,"081C3E7FFFFFFFFF")
42 CALL CHAR(129,"0000000000387CFF")
44 CALL VCHAR(2,21,98,12)
46 CALL VCHAR(2,20,104)
48 CALL VCHAR(3,20,105,9)
50 CALL VCHAR(3,19,104)
52 CALL VCHAR(4,19,105,8)
54 CALL VCHAR(4,18,104)
56 CALL VCHAR(5,18,105,7)
58 CALL VCHAR(5,17,104)
60 CALL VCHAR(6,17,105,6)
62 CALL VCHAR(6,16,104)
64 CALL VCHAR(7,16,105,5)
66 CALL VCHAR(7,15,104)
68 CALL VCHAR(8,15,105,4)
70 CALL VCHAR(8,14,104)
72 CALL VCHAR(9,14,105,3)
74 CALL VCHAR(9,13,104)
76 CALL VCHAR(10,13,105,2)
78 CALL VCHAR(10,12,104)
80 CALL VCHAR(11,12,105)
82 CALL VCHAR(11,11,104)
84 CALL HCHAR(12,11,99,10)
86 CALL HCHAR(14,5,112)
88 CALL HCHAR(14,6,114,19)
90 CALL HCHAR(14,25,113)
92 CALL HCHAR(15,6,112)
94 CALL HCHAR(15,7,114,17)
96 CALL HCHAR(15,24,113)
98 CALL HCHAR(16,7,112)

```

```
100 CALL HCHAR(16,8,114,15)
102 CALL HCHAR(16,23,113)
104 CALL HCHAR(17,8,112)
106 CALL HCHAR(17,9,114,13)
108 CALL HCHAR(17,22,113)
110 CALL HCHAR(18,9,112)
112 CALL HCHAR(18,10,114,11)
114 CALL HCHAR(18,21,113)
116 CALL HCHAR(13,26,100,3)
118 CALL HCHAR(16,9,120)
120 CALL HCHAR(16,12,120)
122 CALL HCHAR(16,15,120)
124 CALL HCHAR(16,18,120)
126 CALL HCHAR(16,21,120)
128 CALL HCHAR(19,1,97,9)
130 CALL HCHAR(19,10,128,11)
132 CALL HCHAR(19,21,97,12)
134 FOR DELAY=1 TO 200
136 NEXT DELAY
138 CALL HCHAR(19,1,101,9)
140 CALL HCHAR(19,10,129,11)
142 CALL HCHAR(19,21,101,12)
144 FOR DELAY=1 TO 200
146 NEXT DELAY
148 GOTO 128
150 END
```

OOPS, DON'T LOOK

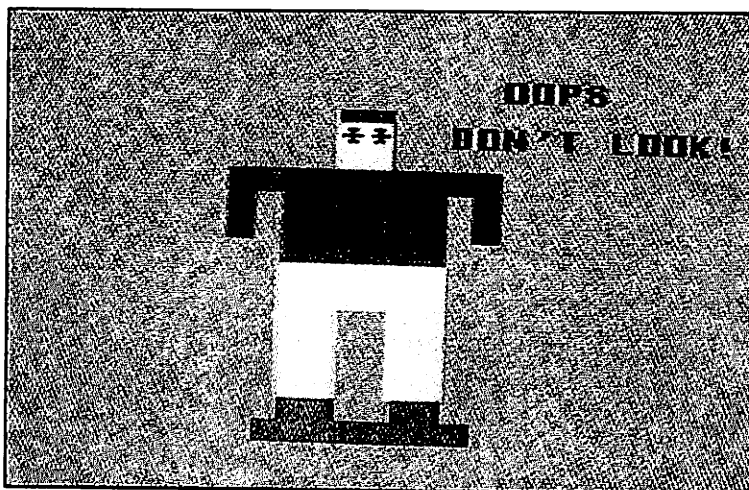
By using CALL HCHAR and CALL VCHAR you can create movement. Watch these commands produce an embarrassing situation!

PROGRAM STRUCTURE

15-30, 60-75	define characters
35-55	set colors
80-230	place man on screen
235-240	time delay
245-295	make the pants fall down
300-355	place words on screen
360-365	time delay
370	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Give the man some hands or add sound as his pants fall. Make the man turn red from embarrassment as his pants fall.



```

5 REM OOPS, DON'T LOOK!
10 CALL CLEAR
15 H$="00000000AAAAAAA"
20 A$="FFFFFFFFFFFFFFFF"
25 B$="FFFFFFFFFFFFFFFF"
30 G$="FFFFFFFFFFFFFFFF"
35 CALL COLOR(16,2,4)
40 CALL COLOR(13,13,13)
45 CALL COLOR(14,10,10)
50 CALL COLOR(2,2,10)
55 CALL COLOR(15,5,5)
60 CALL CHAR(128,A$)
65 CALL CHAR(136,B$)
70 CALL CHAR(144,G$)
75 CALL CHAR(152,H$)
80 CALL HCHAR(6,15,152,2)
85 CALL HCHAR(7,15,42,2)
90 CALL HCHAR(8,15,136,2)
95 CALL VCHAR(9,14,128,4)
100 CALL VCHAR(9,15,128,4)
105 CALL VCHAR(9,16,128,4)
110 CALL VCHAR(9,17,128,4)
115 CALL VCHAR(9,18,128,4)
120 CALL VCHAR(9,13,128,4)
125 CALL VCHAR(9,14,128,4)
130 CALL VCHAR(9,12,128)
135 CALL VCHAR(9,11,128,3)
140 CALL VCHAR(9,19,128)
145 CALL VCHAR(9,20,128,3)

```

```
150 CALL HCHAR(5,21,127,12)
155 CALL HCHAR(7,19,127,12)
160 CALL HCHAR(20,12,127,12)
165 CALL HCHAR(20,13,144,2)
170 CALL HCHAR(20,17,144,2)
175 CALL HCHAR(19,13,144,2)
180 CALL HCHAR(19,17,144,2)
185 CALL HCHAR(18,13,144,2)
190 CALL HCHAR(18,17,144,2)
195 CALL HCHAR(17,13,144,2)
200 CALL HCHAR(17,17,144,2)
205 CALL HCHAR(16,13,144,2)
210 CALL HCHAR(16,17,144,2)
215 CALL HCHAR(15,13,144,2)
220 CALL HCHAR(15,17,144,2)
225 CALL HCHAR(14,13,144,6)
230 CALL HCHAR(13,13,144,6)
235 FOR DELAY=1 TO 500
240 NEXT DELAY
245 CALL HCHAR(13,13,136,6)
250 CALL HCHAR(14,13,136,6)
255 CALL HCHAR(15,13,136,2)
260 CALL HCHAR(15,17,136,2)
265 CALL HCHAR(16,13,136,2)
270 CALL HCHAR(16,17,136,2)
275 CALL HCHAR(17,13,136,2)
280 CALL HCHAR(17,17,136,2)
285 CALL HCHAR(18,13,136,2)
290 CALL HCHAR(18,17,136,2)
295 CALL HCHAR(20,12,144,8)
300 CALL HCHAR(5,21,79,2)
305 CALL HCHAR(5,23,80)
310 CALL HCHAR(5,24,83)
315 CALL HCHAR(7,19,68)
320 CALL HCHAR(7,20,79)
325 CALL HCHAR(7,21,78)
330 CALL HCHAR(7,22,39)
335 CALL HCHAR(7,23,84)
340 CALL HCHAR(7,25,76)
345 CALL HCHAR(7,26,79,2)
350 CALL HCHAR(7,28,75)
355 CALL HCHAR(7,29,33)
360 FOR DELAY=1 TO 500
365 NEXT DELAY
370 GOTO 150
375 END
```

TUNNEL VISION

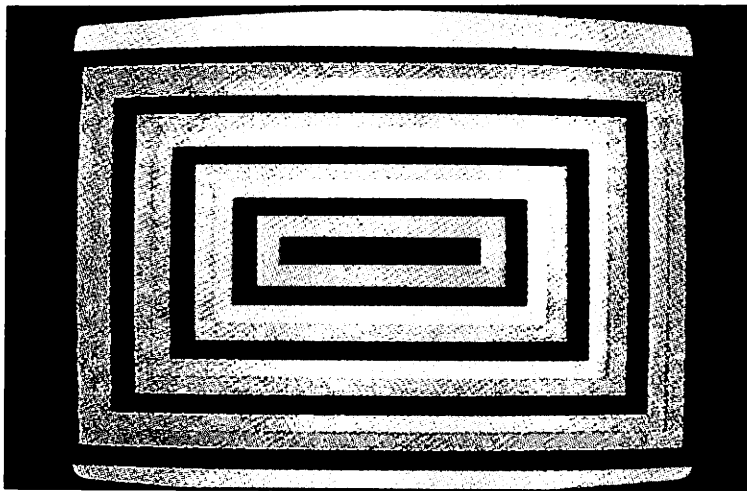
You are about to give the illusion of movement by just changing colors. Watch your screen—you will soon be speeding down a tunnel.

PROGRAM STRUCTURE

15, 60-80	set colors
20-55	define characters
85-330	place tunnel on screen
335-375	contain data to change the colors
380-405	continuously read and change colors
415	keeps the program running

RECOMMENDATIONS FOR ALTERATIONS

Try different color combinations and different speeds.



```

5 REM TUNNEL VISION
10 CALL CLEAR
15 CALL SCREEN(16)
20 CALL CHAR(72,"0000FFFFFFFFFFFF")
25 CALL CHAR(77,"FFFFFFFFFFFF0000")
30 CALL CHAR(73,"00003F3F3F3F3F3F")
35 CALL CHAR(74,"0000FCFCFCFCFCFC")
40 CALL CHAR(75,"FCFCFCFCFCFC0000")
45 CALL CHAR(76,"3F3F3F3F3F3F0000")
50 CALL CHAR(80,"FFFFFFFFFFFFFFFF")
55 CALL CHAR(88,"FFFFFFFFFFFFFFFF")
60 CALL CHAR(97,"FFFFFFFFFFFFFFFF")
65 CALL COLOR(6,2,14)
70 CALL COLOR(7,14,1)
75 CALL COLOR(8,2,1)
80 CALL COLOR(9,9,1)
85 CALL HCHAR(13,12,76)
90 CALL HCHAR(13,21,75)
95 CALL VCHAR(12,12,73)
100 CALL VCHAR(12,21,74)
105 CALL HCHAR(13,13,77,8)
110 CALL HCHAR(12,13,72,8)
115 CALL VCHAR(12,22,88,2)
120 CALL VCHAR(12,11,88,2)
125 CALL HCHAR(14,11,88,12)
130 CALL HCHAR(11,11,88,12)
135 CALL VCHAR(11,10,80,4)
140 CALL VCHAR(11,23,80,4)
145 CALL HCHAR(15,10,80,14)
150 CALL HCHAR(10,10,80,14)
155 CALL VCHAR(10,24,97,6)
160 CALL VCHAR(10,9,97,6)
165 CALL HCHAR(16,9,97,16)
170 CALL HCHAR(9,9,97,16)
175 CALL VCHAR(9,25,88,8)
180 CALL VCHAR(9,8,88,8)
185 CALL HCHAR(17,8,88,18)
190 CALL HCHAR(8,8,88,18)
195 CALL VCHAR(8,26,80,10)
200 CALL VCHAR(8,7,80,10)
205 CALL HCHAR(18,7,80,20)
210 CALL HCHAR(7,7,80,20)
215 CALL VCHAR(7,27,97,12)
220 CALL VCHAR(7,6,97,12)
225 CALL HCHAR(19,6,97,22)
230 CALL HCHAR(6,6,97,22)
235 CALL VCHAR(6,28,88,14)
240 CALL VCHAR(6,5,88,14)
245 CALL HCHAR(20,5,88,24)

```

```
250 CALL HCHAR(5,5,88,24)
255 CALL VCHAR(5,29,80,16)
260 CALL VCHAR(5,4,80,16)
265 CALL HCHAR(21,4,80,26)
270 CALL HCHAR(4,4,80,26)
275 CALL VCHAR(4,30,97,18)
280 CALL VCHAR(4,3,97,18)
285 CALL HCHAR(22,3,97,28)
290 CALL HCHAR(3,3,97,28)
295 CALL VCHAR(3,31,88,20)
300 CALL VCHAR(3,2,88,20)
305 CALL HCHAR(23,2,88,30)
310 CALL HCHAR(2,2,88,30)
315 CALL VCHAR(2,32,80,22)
320 CALL VCHAR(2,1,80,22)
325 CALL HCHAR(24,1,80,32)
330 CALL HCHAR(1,1,80,32)
335 DATA 2,9,14
340 DATA 14,2,9
345 DATA 9,14,2
350 DATA 2,9,14
355 DATA 14,2,9
360 DATA 9,14,2
365 DATA 2,9,14
370 DATA 14,2,9
375 DATA 9,14,2
380 FOR I=1 TO 9
385 READ A,B,C
390 CALL COLOR(9,A,1)
395 CALL COLOR(8,B,1)
400 CALL COLOR(7,C,1)
405 NEXT I
410 RESTORE 335
415 GOTO 335
420 END
```

BLIND AS A BAT

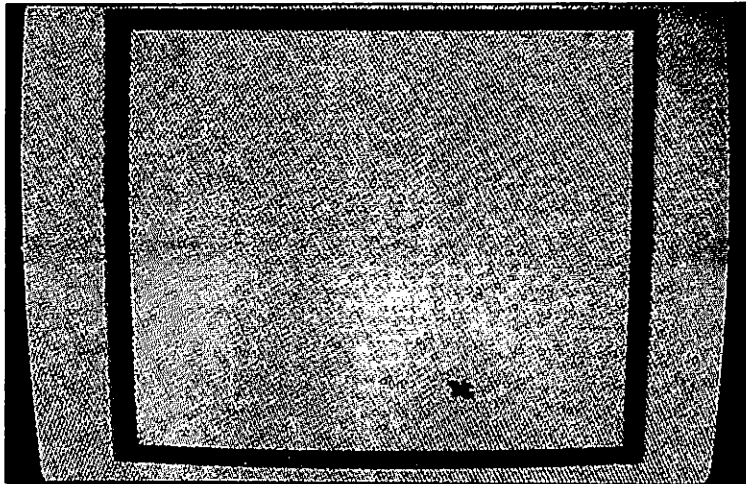
Everyone knows that bats are not blind. But what if one was and it was thrown into a room with no openings? Note that four different characters were defined as the bat—this makes the bat appear to turn in different directions.

PROGRAM STRUCTURE

15-25	set colors
30-50	define characters
55-70	place room on screen
75-100	give starting coordinates for the bat
105-120	are checks for keeping the bat in the room
125-130	time delay
135-150	control bat's movements
160-230	control sound and the turns the bat makes
235	keeps the program running

RECOMMENDATIONS FOR ALTERATIONS

Increase the speed of the bat or enlarge the room.



```
5 REM BLIND AS A BAT
10 CALL CLEAR
15 CALL SCREEN(10)
20 CALL COLOR(9,2,1)
25 CALL COLOR(11,14,1)
30 CALL CHAR(96,"E4E67F3C3C7F2703")
35 CALL CHAR(97,"2767FE3C3CFEE4C0")
40 CALL CHAR(98,"C0E4FE3C3CFE6727")
45 CALL CHAR(99,"03277F3C3C7FE6E4")
50 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
55 CALL HCHAR(1,5,112,24)
60 CALL HCHAR(24,5,112,24)
65 CALL VCHAR(1,5,112,24)
70 CALL VCHAR(1,28,112,24)
75 XOLD=12
80 YOLD=6
85 XDIR=1
90 YDIR=1
95 X=XOLD+XDIR
100 Y=YOLD+YDIR
105 IF X<2 THEN 160
110 IF X>23 THEN 180
115 IF Y<6 THEN 200
120 IF Y>27 THEN 220
125 FOR I=1 TO 20
130 NEXT I
135 CALL HCHAR(XOLD,YOLD,32)
140 CALL HCHAR(X,Y,Z)
145 XOLD=X
```

150 YOLD=Y
155 GOTO 95
160 XDIR=-XDIR
165 CALL SOUND(50,-1,2)
170 Z=98
175 GOTO 95
180 XDIR=-XDIR
185 CALL SOUND(50,-1,2)
190 Z=96
195 GOTO 95
200 YDIR=-YDIR
205 CALL SOUND(50,-1,2)
210 Z=99
215 GOTO 95
220 YDIR=-YDIR
225 CALL SOUND(50,-1,2)
230 Z=97
235 GOTO 95
240 END

DANGER IN THE DEEP

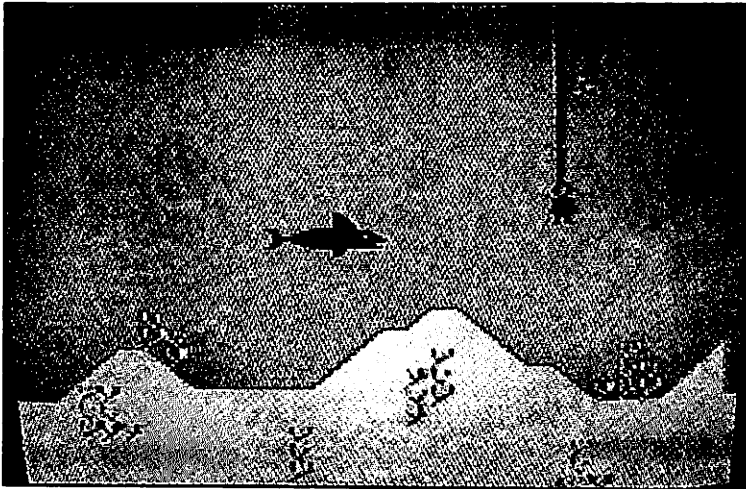
In this program we combine a little movement with sound. You can watch the bubbles rise from the diver's helmet while he watches something more important.

PROGRAM STRUCTURE

6-18	set colors
20-54	define characters
58-174	place characters on screen
176-196	create movement and sound for bubbles
198	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Put a crab on the ocean floor or some smaller fish in; fill the screen with other sea creatures.



```

2 REM    DANGER IN THE DEEP
4 CALL CLEAR
6 CALL SCREEN(6)
8 CALL COLOR(3,15,6)
10 CALL COLOR(9,2,1)
12 CALL COLOR(10,2,1)
14 CALL COLOR(11,11,1)
16 CALL COLOR(12,3,11)
18 CALL COLOR(2,12,1)
20 CALL CHAR(96,"1010101010101010")
22 CALL CHAR(120,"BABABA4C302040B0")
24 CALL CHAR(121,"B0B4B4B8B4B3010BA")
26 CALL CHAR(97,"1824424224187E7E")
28 CALL CHAR(40,"BABABA4C302040B0")
30 CALL CHAR(41,"B0B4B4B8B4B3010BA")
32 CALL CHAR(98,"7E7EBDBD3C242466")
34 CALL CHAR(99,"70180C0F0F0C1870")
36 CALL CHAR(100,"00073FFFFFF3F0700")
38 CALL CHAR(101,"FFFFFFFFFFFFFFFF")
40 CALL CHAR(102,"FCFFFFFFFFFFFFFFFF")
42 CALL CHAR(103,"00E0F89EFFFFFF0FE")
44 CALL CHAR(104,"0000B0C0E0F0F8FC")
46 CALL CHAR(105,"E0C0B0")
48 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
50 CALL CHAR(113,"0103070F1F3F7FFF")
52 CALL CHAR(114,"B0C0E0F0F8FCFEFF")
54 CALL CHAR(48,"207020040E44E040")
56 XOLD=8
58 CALL VCHAR(1,24,96,8)
60 CALL HCHAR(9,24,97)
62 CALL HCHAR(10,24,98)
64 CALL HCHAR(11,12,99)
66 CALL HCHAR(11,13,100)
68 CALL HCHAR(11,14,101)
70 CALL HCHAR(11,15,102)
72 CALL HCHAR(11,16,103)
74 CALL HCHAR(10,15,104)
76 CALL HCHAR(12,15,105)
78 CALL HCHAR(17,5,113)
80 CALL HCHAR(17,6,112)
82 CALL HCHAR(17,7,114)
84 CALL HCHAR(18,4,113)
86 CALL HCHAR(18,5,112,3)
88 CALL HCHAR(18,8,114)
90 CALL HCHAR(19,3,113)
92 CALL HCHAR(19,4,112,21)
94 CALL HCHAR(18,14,113)
96 CALL HCHAR(18,15,112,9)
98 CALL HCHAR(18,24,114)
100 CALL HCHAR(17,15,113)

```

```

102 CALL HCHAR(17,16,112,6)
104 CALL HCHAR(17,22,114)
106 CALL HCHAR(16,16,113)
108 CALL HCHAR(16,17,112,4)
110 CALL HCHAR(16,21,114)
112 CALL HCHAR(15,18,113)
114 CALL HCHAR(15,19,112)
116 CALL HCHAR(15,20,114)
118 CALL HCHAR(19,25,114)
120 CALL HCHAR(17,31,113)
122 CALL HCHAR(18,30,113)
124 CALL HCHAR(18,31,112)
126 CALL HCHAR(19,29,113)
128 CALL HCHAR(19,30,112,2)
130 CALL HCHAR(20,1,112,160)
132 CALL HCHAR(17,19,120)
134 CALL HCHAR(18,18,120)
136 CALL HCHAR(18,19,121)
138 CALL HCHAR(19,18,121)
140 CALL HCHAR(19,19,120)
142 CALL HCHAR(20,18,120)
144 CALL HCHAR(19,4,120,2)
146 CALL HCHAR(21,13,120)
148 CALL HCHAR(22,13,121)
150 CALL HCHAR(23,13,120)
152 CALL HCHAR(24,24,120,3)
154 CALL HCHAR(23,25,121)
156 CALL HCHAR(22,25,120)
158 CALL HCHAR(20,4,121,2)
160 CALL HCHAR(17,8,40)
162 CALL HCHAR(16,7,40)
164 CALL HCHAR(16,8,41)
166 CALL HCHAR(15,7,41)
168 CALL HCHAR(19,26,40,3)
170 CALL HCHAR(18,27,41,2)
172 CALL HCHAR(17,27,40)
174 CALL HCHAR(21,4,120,3)
176 CALL HCHAR(XOLD,25,48)
178 CALL SOUND(400,-4,2)
180 FOR DELAY=1 TO 100
182 NEXT DELAY
184 X=XOLD-1
186 IF X<1 THEN 188 ELSE 192
188 X=8
190 CALL SOUND(400,-4,2)
192 CALL HCHAR(X,25,48)
194 CALL HCHAR(XOLD,25,32)
196 XOLD=X
198 GOTO 180
200 END

```

I SAW THE LIGHT

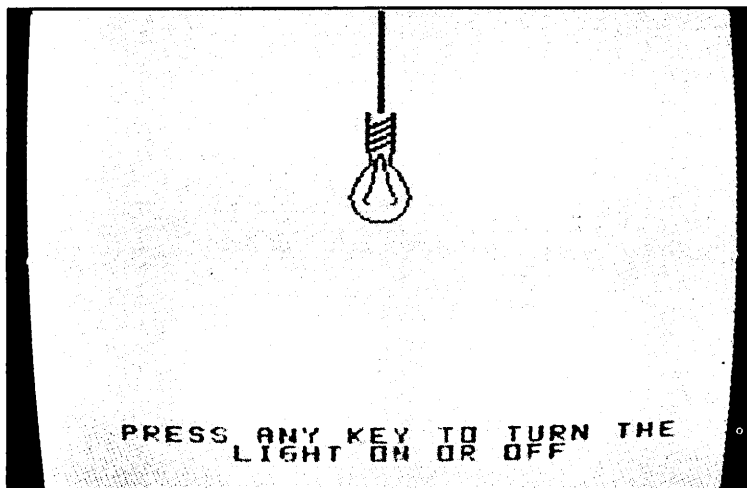
Ever been left in the dark? Well, now you too can see the light just by pressing any key.

PROGRAM STRUCTURE

40-200	define characters
230-320	set colors
330-490	place light bulb on screen
500-510	waits for a key to be pressed
520-610	reset colors
620-630	waits for a key to be pressed
640	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Place other objects in the room. Set color for the room so that when the light is turned on it'll have a different "mood."




```

10 REM      I SAW THE LIGHT
20 CALL CLEAR
30 CALL SCREEN(2)
40 CALL CHAR(120,"1818181818181818")
50 CALL CHAR(121,"0000000000010204")
60 CALL CHAR(122,"9999A5A5A5242442")
70 CALL CHAR(123,"0000000000B04020")
80 CALL CHAR(124,"0408081010202021")
90 CALL CHAR(125,"42424242818181")
100 CALL CHAR(126,"2010100B0B040484")
110 CALL CHAR(127,"414222212010100B")
120 CALL CHAR(128,"00245A81")
130 CALL CHAR(96,"0101010101010101")
140 CALL CHAR(97,"B0B0B0B0B0B0B0B0")
150 CALL CHAR(104,"0000030C30C30C30")
160 CALL CHAR(105,"C30C30C30C30C000")
170 CALL CHAR(112,"82424484040B0B10")
180 CALL CHAR(113,"0403")
190 CALL CHAR(114,"0000FF")
200 CALL CHAR(115,"20C0")
210 PRINT "  PRESS ANY KEY TO TURN THE"
220 PRINT "      LIGHT ON OR OFF"
230 CALL SCREEN(2)
240 CALL COLOR(10,2,2)
250 CALL COLOR(9,2,2)
260 CALL COLOR(11,2,2)
270 CALL COLOR(12,2,2)
280 CALL COLOR(13,2,2)
290 CALL COLOR(5,16,1)
300 CALL COLOR(6,16,1)
310 CALL COLOR(7,16,1)
320 CALL COLOR(8,16,1)
330 CALL VCHAR(1,16,120,5)
340 CALL VCHAR(6,15,96,2)
350 CALL VCHAR(6,16,104)
360 CALL VCHAR(6,17,97,2)
370 CALL VCHAR(7,16,105)
380 CALL HCHAR(8,15,121)
390 CALL HCHAR(8,16,122)
400 CALL HCHAR(8,17,123)
410 CALL HCHAR(9,15,124)
420 CALL HCHAR(9,16,125)
430 CALL HCHAR(9,17,126)
440 CALL HCHAR(10,15,127)
450 CALL HCHAR(10,16,128)
460 CALL HCHAR(10,17,112)
470 CALL HCHAR(11,15,113)
480 CALL HCHAR(11,16,114)
490 CALL HCHAR(11,17,115)

```

```
500 CALL KEY(0,A,STATUS)
510 IF STATUS=0 THEN 500
520 CALL COLOR(13,9,1)
530 CALL COLOR(9,2,1)
540 CALL COLOR(10,2,11)
550 CALL COLOR(11,2,1)
560 CALL COLOR(12,2,1)
570 CALL SCREEN(15)
580 CALL COLOR(5,2,1)
590 CALL COLOR(6,2,1)
600 CALL COLOR(7,2,1)
610 CALL COLOR(8,2,1)
620 CALL KEY(0,B,STATUS)
630 IF STATUS=0 THEN 620
640 GOTO 230
650 END
```

UP, UP, AND AWAY

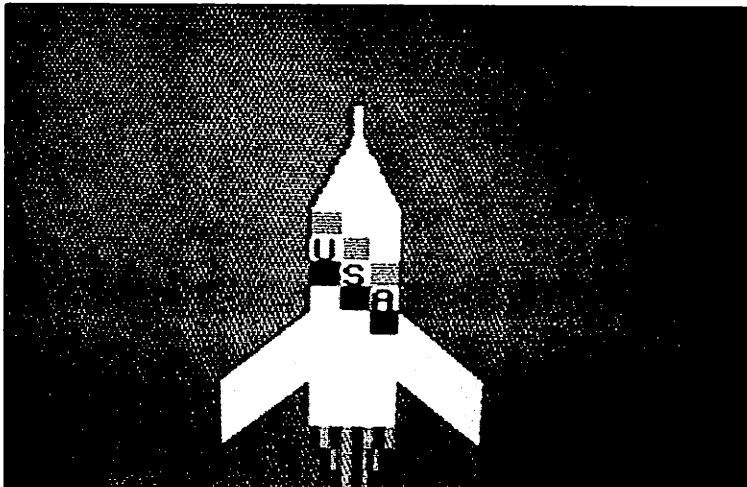
Let's blast off with this program, which creates movement by continuously printing a blank line. Add some sound and you complete the illusion.

PROGRAM STRUCTURE

6-18	set colors
20-50	define characters
52-54	time delay
56-130	place rocket on screen
132-134	time delay before rocket's engines fire
136-144	create flames and sound
146-154	move the rocket off the screen
156	keeps the program running

RECOMMENDATIONS FOR ALTERATIONS

Display a countdown on the screen. To do this, you may have to change the time delay.



```

2 REM  UP, UP, AND AWAY
4 CALL CLEAR
6 CALL SCREEN(6)
8 CALL COLOR(9,16,1)
10 CALL COLOR(10,10,6)
12 CALL COLOR(11,5,1)
14 CALL COLOR(5,2,16)
16 CALL COLOR(7,2,16)
18 CALL COLOR(12,1,16)
20 CALL CHAR(97,"1818181818181818")
22 CALL CHAR(98,"18183C3C3C7E7EFF")
24 CALL CHAR(99,"010103030307070F")
26 CALL CHAR(100,"8080C0C0C0E0E0F0")
28 CALL CHAR(101,"0F1F1F3F3F7F7FFF")
30 CALL CHAR(102,"F0F8F8FCFCFEFEFF")
32 CALL CHAR(103,"FFFFFFFFFFFFFFFF")
34 CALL CHAR(104,"FFFFFFFFFFFFFFFF")
36 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
38 CALL CHAR(120,"0103070F1F3F7FFF")
40 CALL CHAR(121,"FFFEFCF8F0E0C080")
42 CALL CHAR(122,"FF7F3F1F0F070301")
44 CALL CHAR(124,"80C0E0F0F8FCFEFF")
46 CALL CHAR(105,"5555555555555555")
48 CALL CHAR(106,"0A0A0A0A0A0A0A0A")
50 CALL CHAR(107,"A0A0A0A0A0A0A0A0")
52 FOR DELAY=1 TO 600
54 NEXT DELAY
56 CALL VCHAR(9,17,97)
58 CALL VCHAR(10,17,98)
60 CALL VCHAR(11,16,99)
62 CALL VCHAR(11,17,103,3)
64 CALL VCHAR(11,18,100)
66 CALL VCHAR(12,16,101)
68 CALL VCHAR(12,18,102)
70 CALL VCHAR(13,16,104)
72 CALL VCHAR(13,18,103,2)
74 CALL VCHAR(14,16,85)
76 CALL VCHAR(14,17,104)
78 CALL VCHAR(15,16,112)
80 CALL VCHAR(15,17,83)
82 CALL VCHAR(15,18,104)
84 CALL VCHAR(16,16,103,6)
86 CALL VCHAR(16,17,112)
88 CALL VCHAR(16,18,65)
90 CALL VCHAR(17,17,103,5)
92 CALL VCHAR(17,18,112)
94 CALL VCHAR(18,18,103,4)
96 CALL VCHAR(17,15,121)
98 CALL VCHAR(18,14,121)

```

```

100 CALL VCHAR(18,15,103,2)
102 CALL VCHAR(19,13,121)
104 CALL VCHAR(19,14,103,2)
106 CALL VCHAR(20,13,103,2)
108 CALL VCHAR(20,15,120)
110 CALL VCHAR(21,14,120)
112 CALL VCHAR(22,13,120)
114 CALL VCHAR(17,19,122)
116 CALL VCHAR(18,19,103,2)
118 CALL VCHAR(18,20,122)
120 CALL VCHAR(19,20,103,2)
122 CALL VCHAR(19,21,122)
124 CALL VCHAR(20,21,103,2)
126 CALL VCHAR(20,19,124)
128 CALL VCHAR(21,20,124)
130 CALL VCHAR(22,21,124)
132 FOR DELAY=1 TO 1000
134 NEXT DELAY
136 CALL HCHAR(22,16,105,3)
138 CALL HCHAR(23,16,106)
140 CALL VCHAR(23,17,105,2)
142 CALL VCHAR(23,18,107)
144 CALL SOUND(4250,-5,1)
146 FOR J=1 TO 24
148 FOR I=1 TO 42
150 NEXT I
152 PRINT
154 NEXT J
156 GOTO 52
158 END

```

SPY IN THE SKY

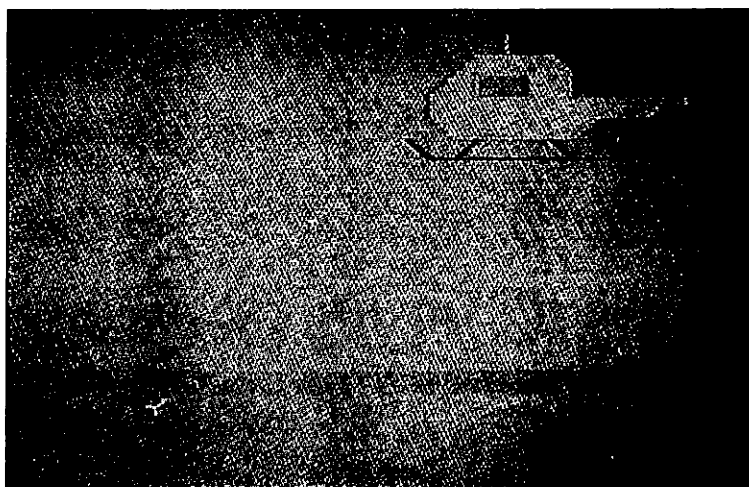
In this program you can use two different methods for moving objects: You can continuously print blank lines or you can change coordinates of an object. This is done by using a variable, such as `xold`, and calling it `x`. By doing that, you give `xold` a new value.

PROGRAM STRUCTURE

6-36	define characters
38-46	set colors
48-90	place helicopter on screen
92-94, 116-118	time delays
98-108	control the helicopter's sound and movement
110-134	control missile's movement
136-176	create explosion
178-180, 184-186	time delays
188	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

After the explosion, have a man parachute to safety. Have several missiles approach the helicopter.



```
2 REM  SPY IN THE SKY
4 CALL CLEAR
6 CALL CHAR(96,"FFFFFFFFFFFFFFFF")
8 CALL CHAR(97,"0103070F1F3F7FFF")
10 CALL CHAR(98,"80C0E0F0F8FCFEFF")
12 CALL CHAR(99,"FFFFFFFFCF8F0C0")
14 CALL CHAR(100,"FFFFFF")
16 CALL CHAR(101,"FF7F3F1F0F070301")
18 CALL CHAR(102,"FFFEFCF8F0E0C080")
20 CALL CHAR(104,"000000000000FFFF")
22 CALL CHAR(105,"1818181818181818")
24 CALL CHAR(106,"01020408102040FF")
26 CALL CHAR(107,"80402010080402FF")
28 CALL CHAR(108,"00000000000000FF")
30 CALL CHAR(109,"8040201008040201")
32 CALL CHAR(112,"03070EFCF8181818")
34 CALL CHAR(40,"3C7EFFFFFFFF7E3C")
36 CALL CHAR(41,"5555AAAA5555AAAA")
38 CALL SCREEN(6)
40 CALL COLOR(2,10,1)
42 CALL COLOR(9,4,1)
44 CALL COLOR(10,2,1)
46 CALL COLOR(11,15,1)
48 CALL HCHAR(17,18,104,9)
50 CALL HCHAR(18,22,105)
52 CALL HCHAR(19,20,97)
54 CALL HCHAR(19,21,96,3)
56 CALL HCHAR(19,24,98)
58 CALL HCHAR(20,19,97)
```

```

60 CALL HCHAR(20,20,96)
62 CALL HCHAR(20,23,96,2)
64 CALL HCHAR(21,19,96,9)
66 CALL HCHAR(22,19,101)
68 CALL HCHAR(22,20,96,5)
70 CALL HCHAR(22,25,102)
72 CALL HCHAR(21,28,99)
74 CALL HCHAR(21,29,100)
76 CALL HCHAR(20,29,105)
78 CALL HCHAR(19,28,104,3)
80 CALL HCHAR(23,18,109)
82 CALL HCHAR(23,19,108)
84 CALL HCHAR(23,20,106)
86 CALL HCHAR(23,21,108,3)
88 CALL HCHAR(23,24,107)
90 CALL HCHAR(23,25,108,2)
92 FOR DELAY=1 TO 750
94 NEXT DELAY
96 CALL SOUND(4250,-4,3)
98 FOR J=1 TO 16
100 FOR I=1 TO 55
102 NEXT I
104 PRINT
106 NEXT J
108 CALL SOUND(4250,-4,3)
110 XOLD=24
112 YOLD=3
114 CALL HCHAR(XOLD,YOLD,112)
116 FOR F=1 TO 30
118 NEXT F
120 X=XOLD-1
122 Y=YOLD+1
124 IF X<6 THEN 136
126 CALL HCHAR(XOLD,YOLD,32)
128 CALL HCHAR(X,Y,112)
130 XOLD=X
132 YOLD=Y
134 GOTO 116
136 CALL SOUND(700,-7,1)
138 CALL CLEAR
140 CALL SCREEN(6)
142 CALL HCHAR(5,21,41,3)
144 CALL HCHAR(4,21,41,3)
146 CALL HCHAR(6,21,41,3)
148 CALL HCHAR(5,24,40,3)
150 CALL HCHAR(5,18,40,3)
152 CALL VCHAR(7,22,40,3)
154 FOR B=1 TO 100

```



```
156 NEXT B
158 CALL HCHAR(3,20,41)
160 CALL HCHAR(3,24,41)
162 CALL HCHAR(7,21,41)
164 CALL HCHAR(7,24,41)
166 CALL HCHAR(2,19,41)
168 CALL HCHAR(2,25,41)
170 CALL HCHAR(8,19,41)
172 CALL HCHAR(8,26,41)
174 CALL HCHAR(9,18,41)
176 CALL HCHAR(9,27,41)
178 FOR A=1 TO 80
180 NEXT A
182 CALL CLEAR
184 FOR E=1 TO 500
186 NEXT E
188 GOTO 38
190 END
```

OUT OF THIS WORLD

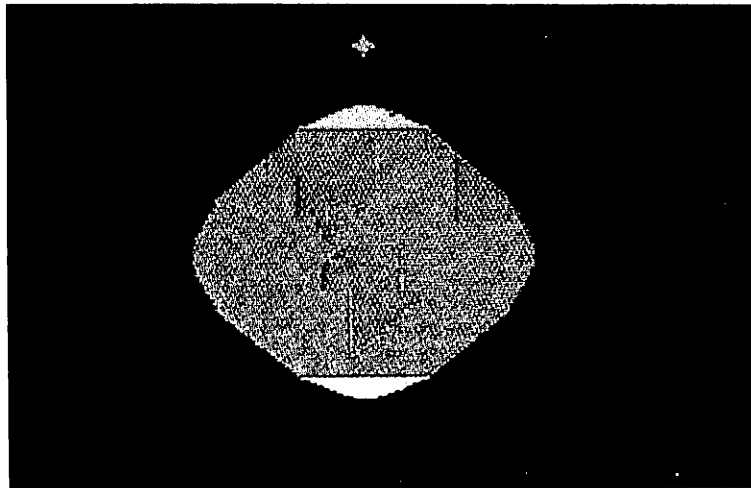
Now that we have both men and women in outer space, there is no reason why the rest of us can't be out there too. Let's go!

PROGRAM STRUCTURE

30-100	set colors
110-450	define characters
460-1080	place Earth on screen
1090-1320	control movement of satellite
1330	keeps program running
1340-1370	are a subroutine to control sound and time delay

RECOMMENDATIONS FOR ALTERATIONS

Put two satellites in orbit and have them dock. Add the moon and other heavenly bodies for background.



```

10 REM    OUT OF THIS WORLD
20 CALL CLEAR
30 CALL COLOR(3,15,1)
40 CALL COLOR(4,5,1)
50 CALL COLOR(5,5,1)
60 CALL COLOR(6,13,1)
70 CALL COLOR(7,13,5)
80 CALL COLOR(8,13,5)
90 CALL COLOR(12,15,2)
100 CALL SCREEN(2)
110 CALL CHAR(48,"18183CFFFF3C1818")
120 CALL CHAR(51,"FFFFFFFFFFFFFFFF")
130 CALL CHAR(56,"010101030307070F")
140 CALL CHAR(57,"0F1F1F3F3F7F7F7F")
150 CALL CHAR(58,"7F7F7F3F3F1F1F0F")
160 CALL CHAR(59,"0F07070303010101")
170 CALL CHAR(60,"FFFFFFFFFFFFFFFF")
180 CALL CHAR(61,"B0B0B0C0C0E0E0F0")
190 CALL CHAR(62,"F0F8F8FCFCFEFEFE")
200 CALL CHAR(63,"FEFEFEFCFCF8F8F0")
210 CALL CHAR(64,"F0E0E0C0C0B0B0B0")
220 CALL CHAR(65,"FFFEFCF8F0E0C0B0")
230 CALL CHAR(66,"FF7F3F1F0F070301")
240 CALL CHAR(67,"0103070F1F3F7FFF")
250 CALL CHAR(80,"FFFFEFE7E7E7B3B3")
260 CALL CHAR(81,"F0E0C0C0C0C0C0C0")
270 CALL CHAR(82,"03030101")
280 CALL CHAR(83,"C0E0E0E0F0FB7C1F")
290 CALL CHAR(75,"B0C0E0F0F8FCFEFF")
300 CALL CHAR(76,"0103070F1F3F7FFF")
310 CALL CHAR(77,"FFFFFFFFFFFFFFFF")
320 CALL CHAR(84,"E0E0703B3B18")
330 CALL CHAR(85,"F8F0E0E0E0E0F0F0")
340 CALL CHAR(86,"FFFEFEFEFEFEFEFC")
350 CALL CHAR(87,"00000C3D3DFFFFFF")
360 CALL CHAR(88,"FFFEFCF8F0E0C0B0")
370 CALL CHAR(89,"0103070F1F3F7FFF")
380 CALL CHAR(120,"00000000B0E0FBFF")
390 CALL CHAR(121,"FF1F0701")
400 CALL CHAR(122,"FFFFFFFF7F1F07")
410 CALL CHAR(123,"FFFFFFFFFFFF8E")
420 CALL CHAR(124,"FFF8E08")
430 CALL CHAR(125,"0000000001071FFF")
440 CALL CHAR(126,"00071F7FFFFFFFFF")
450 CALL CHAR(127,"00E0FBF7FFFFFFFF")
460 CALL HCHAR(6,14,125)
470 CALL HCHAR(6,15,126)
480 CALL HCHAR(6,16,51)
490 CALL HCHAR(6,17,127)

```

500 CALL HCHAR(6,18,120)
510 CALL HCHAR(9,21,75)
520 CALL HCHAR(8,20,75)
530 CALL HCHAR(7,19,75)
540 CALL HCHAR(10,22,61)
550 CALL HCHAR(11,22,62)
560 CALL HCHAR(12,22,60)
570 CALL HCHAR(13,22,63)
580 CALL HCHAR(14,22,64)
590 CALL HCHAR(10,10,56)
600 CALL HCHAR(11,10,57)
610 CALL HCHAR(12,10,60)
620 CALL HCHAR(13,10,58)
630 CALL HCHAR(14,10,59)
640 CALL HCHAR(18,14,121)
650 CALL HCHAR(18,15,122)
660 CALL HCHAR(18,16,51)
670 CALL HCHAR(18,17,123)
680 CALL HCHAR(18,18,124)
690 CALL HCHAR(17,19,65)
700 CALL HCHAR(16,20,65)
710 CALL HCHAR(15,21,65)
720 CALL HCHAR(17,13,66)
730 CALL HCHAR(17,14,60,5)
740 CALL HCHAR(16,13,60,7)
750 CALL HCHAR(15,12,60,9)
760 CALL HCHAR(14,11,60,11)
770 CALL HCHAR(13,11,60,11)
780 CALL HCHAR(12,11,60,11)
790 CALL HCHAR(11,11,60,11)
800 CALL HCHAR(10,11,60,11)
810 CALL HCHAR(9,12,60,9)
820 CALL HCHAR(8,13,60,7)
830 CALL HCHAR(7,14,60,5)
840 CALL HCHAR(16,12,66)
850 CALL HCHAR(15,11,66)
860 CALL HCHAR(9,11,67)
870 CALL HCHAR(8,12,67)
880 CALL HCHAR(10,14,80)
890 CALL HCHAR(10,15,81)
900 CALL HCHAR(11,14,82)
910 CALL HCHAR(11,15,83)
920 CALL HCHAR(10,16,84)
930 CALL HCHAR(9,16,85)
940 CALL HCHAR(8,16,86)
950 CALL HCHAR(12,17,86)
960 CALL HCHAR(7,16,87)
970 CALL HCHAR(14,18,88)
980 CALL HCHAR(15,17,88)

```
990 CALL HCHAR(16,16,88)
1000 CALL HCHAR(12,15,89)
1010 CALL HCHAR(7,13,76)
1020 CALL HCHAR(8,13,77,4)
1030 CALL HCHAR(9,14,77,3)
1040 CALL HCHAR(9,20,77)
1050 CALL HCHAR(10,20,77,2)
1060 CALL VCHAR(13,15,77)
1070 CALL VCHAR(12,16,77,4)
1080 CALL VCHAR(13,17,77,2)
1090 FOR X=7 TO 18 STEP 5
1100 CALL HCHAR(X,7,48)
1110 GOSUB 1340
1120 CALL HCHAR(X,6,32,3)
1130 GOSUB 1340
1140 NEXT X
1150 FOR Y=11 TO 21 STEP 5
1160 CALL HCHAR(21,Y,48)
1170 GOSUB 1340
1180 CALL HCHAR(21,Y-1,32,3)
1190 GOSUB 1340
1200 NEXT Y
1210 FOR X=18 TO 7 STEP -5
1220 CALL HCHAR(X,25,48)
1230 GOSUB 1340
1240 CALL VCHAR(X-1,25,32,3)
1250 GOSUB 1340
1260 NEXT X
1270 FOR Y=21 TO 11 STEP -5
1280 CALL HCHAR(3,Y,48)
1290 GOSUB 1340
1300 CALL HCHAR(3,Y-1,32,3)
1310 GOSUB 1340
1320 NEXT Y
1330 GOTO 1090
1340 CALL SOUND(50,800,1)
1350 FOR DELAY=1 TO 80
1360 NEXT DELAY
1370 RETURN
1380 END
```

JUST A LITTLE SQUIRT

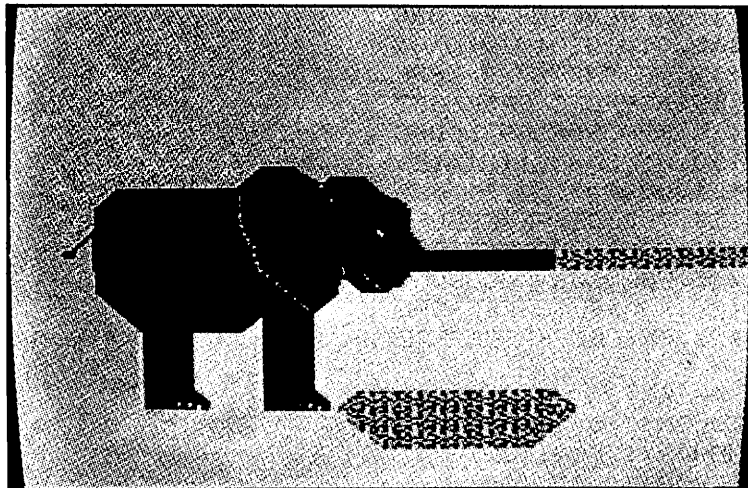
Here we are again, using just CALL HCHAR and CALL VCHAR to create movement. The characters in this program are better defined than the ones in previous programs. The elephant is just having fun—don't get wet!

PROGRAM STRUCTURE

30–430	define characters
440–450	set colors
460–1060	place elephant on screen
1070–1210	control movement and sound

RECOMMENDATIONS FOR ALTERATIONS

Have the water splash off the edge of the screen.



```

10 REM JUST A LITTLE SQUIRT
20 CALL CLEAR
30 READ A
40 IF A=-1 THEN 440
50 READ A$
60 CALL CHAR(A,A$)
70 GOTO 30
80 DATA 40,0101070A092648AD
90 DATA 41,9519071209040101
100 DATA 42,9A19B96A95A89B95
110 DATA 43,8040608038046555
120 DATA 44,555BB03408D0A0C0
130 DATA 97,FFFEFCF8F0E0C0B0
140 DATA 98,FF7F3F1F0F070301
150 DATA 99,0103070F1F3F7FFF
160 DATA 100,FFFFFFFFFFFFFFFF
170 DATA 101,FFFFFFFFFECECF
180 DATA 102,80C0E0F0F8F89898
190 DATA 103,000000000387DFCF
200 DATA 104,00000000FFFFFFFF
210 DATA 105,0000000080C0E0F0
220 DATA 106,EFEFF7FBFDFDFEFE
230 DATA 107,F8FCFEFFFFFFFF
240 DATA 108,0000080F0F8FEFE
250 DATA 109,7F7F7F7F7F7F7FFF
260 DATA 110,F9F8FCFFFFFFFF
270 DATA 111,DFDFDFDFDFDFDFDF
280 DATA 112,008080C0E0E0F0F8
290 DATA 113,DFDFFF8F87030100
300 DATA 114,BFBFDFEFF7FBDFE
310 DATA 115,FFFFFEFCFCF8C0B0
320 DATA 116,0101010101010101
330 DATA 117,7F7F7F7F7F7F3FBF
340 DATA 118,BF9FDFCFEFEFF3F7
350 DATA 119,F7F3FBFBF9FDFDFC
360 DATA 120,0103050911214181
370 DATA 121,070F07
380 DATA 123,7FBFDFEFF7FBDFE
390 DATA 124,C0E0F0FCFEFFFFFF
400 DATA 125,F0F0E0E0C0C0B000
410 DATA 126,FEFFFFFFFFFFFFFFE
420 DATA 127,FFFFFFFFFFFFFF7E
430 DATA -1
440 CALL SCREEN(15)
450 CALL COLOR(2,5,1)
460 CALL HCHAR(12,18,112)
470 CALL HCHAR(10,17,108)
480 CALL VCHAR(11,17,100,3)
490 CALL HCHAR(14,17,115)

```

500 CALL VCHAR(10,16,107)
510 CALL HCHAR(11,16,100,2)
520 CALL VCHAR(12,16,110)
530 CALL VCHAR(13,16,100)
540 CALL VCHAR(14,16,114)
550 CALL VCHAR(9,15,104)
560 CALL VCHAR(9,16,105)
570 CALL VCHAR(10,15,100)
580 CALL VCHAR(11,15,109)
590 CALL VCHAR(12,15,100)
600 CALL VCHAR(13,15,111)
610 CALL VCHAR(14,15,113)
620 CALL VCHAR(9,14,103)
630 CALL VCHAR(10,14,106)
640 CALL VCHAR(11,14,100,4)
650 CALL VCHAR(15,14,97)
660 CALL VCHAR(9,13,124)
670 CALL VCHAR(10,13,100,5)
680 CALL VCHAR(16,13,100,4)
690 CALL VCHAR(20,13,101)
700 CALL VCHAR(20,14,102)
710 CALL VCHAR(9,12,100,5)
720 CALL VCHAR(14,12,123)
730 CALL VCHAR(15,13,123)
740 CALL VCHAR(15,12,100,6)
750 CALL VCHAR(9,11,99)
760 CALL VCHAR(10,11,109)
770 CALL VCHAR(11,11,117)
780 CALL VCHAR(12,11,118)
790 CALL VCHAR(13,11,119)
800 CALL VCHAR(14,11,100,2)
810 CALL VCHAR(16,11,97)
820 CALL VCHAR(10,10,100,7)
830 CALL VCHAR(10,9,100,7)
840 CALL VCHAR(10,8,100,10)
850 CALL VCHAR(20,8,101)
860 CALL VCHAR(20,9,102)
870 CALL VCHAR(10,7,100,11)
880 CALL VCHAR(10,6,100,6)
890 CALL VCHAR(16,6,98)
900 CALL VCHAR(10,5,99)
910 CALL VCHAR(11,5,100,4)
920 CALL VCHAR(15,5,98)
930 CALL VCHAR(13,4,116)
940 CALL HCHAR(12,4,120)
950 CALL HCHAR(13,3,121)
960 CALL VCHAR(13,18,125)
970 CALL VCHAR(14,17,127,6)
980 CALL VCHAR(20,15,40)
990 CALL VCHAR(21,15,41)


```
1000 CALL HCHAR(20,16,42,8)
1010 CALL HCHAR(21,16,42,8)
1020 CALL HCHAR(22,16,41)
1030 CALL HCHAR(22,17,42,6)
1040 CALL HCHAR(20,24,43)
1050 CALL HCHAR(21,24,44)
1060 CALL HCHAR(22,23,44)
1070 CALL VCHAR(13,18,125)
1080 CALL VCHAR(14,17,127,6)
1090 FOR DELAY=1 TO 500
1100 NEXT DELAY
1110 CALL VCHAR(14,17,32,6)
1120 CALL HCHAR(14,17,115)
1130 CALL HCHAR(13,18,126,6)
1140 CALL HCHAR(13,24,42,9)
1150 CALL SOUND(1100,-6,15)
1160 FOR DELAY=1 TO 300
1170 NEXT DELAY
1180 CALL HCHAR(13,24,32,9)
1190 CALL HCHAR(13,18,32,6)
1200 GOTO 1070
1210 END
```

THERE SHE BLOWS

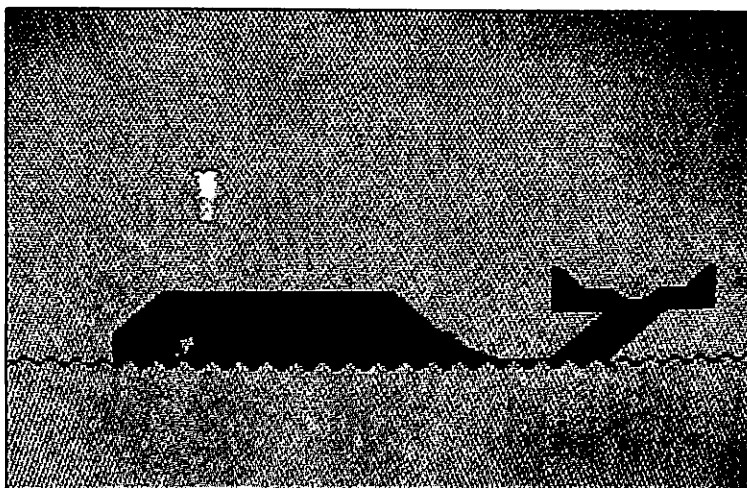
Although this program is short, there are three different movements involved in the running of it. A whale is placed on the screen. He winks his eye and spouts water while waves rise and fall.

PROGRAM STRUCTURE

30-70	set colors
80-240	define characters
250-500	place whale and waves on screen
510-520	time delay
530-600	control spouting of water
610-640	change the waves
650-660	time delay
670	keeps the program running

RECOMMENDATIONS FOR ALTERATIONS

Add sound when the water spouts or movement to the whale's tail.



```
10 REM   THERE SHE BLOWS
20 CALL CLEAR
30 CALL COLOR(2,2,5)
40 CALL COLOR(3,2,5)
50 CALL COLOR(4,5,2)
60 CALL COLOR(5,16,6)
70 CALL SCREEN(5)
80 CALL CHAR(40,"08144180")
90 CALL CHAR(41,"0000000000384483")
100 CALL CHAR(43,"FFFFFFFFFFFFFFFF")
110 CALL CHAR(44,"80C0E0F0F8FCFEFF")
120 CALL CHAR(45,"0103070F1F3F7FFF")
130 CALL CHAR(46,"80C0E0F0F8FCFFFF")
140 CALL CHAR(47,"000000000000C0E0")
150 CALL CHAR(48,"E0F0F8FCFFFFFFFF")
160 CALL CHAR(49,"0000000000E0F0FE")
170 CALL CHAR(50,"FFFEFCF8F0E0C080")
180 CALL CHAR(51,"80C0E0F8FFFFFFFF")
190 CALL CHAR(52,"0103071FFFFFFFFF")
200 CALL CHAR(56,"081C3E7FFFFFFFFF")
210 CALL CHAR(57,"0000000000387CFF")
220 CALL CHAR(58,"071F2F2E4E4C7")
230 CALL CHAR(59,"0000030307F7F4FF")
240 CALL CHAR(64,"4C14283C142C183C")
250 CALL HCHAR(15,8,43,12)
260 CALL HCHAR(14,9,43,10)
270 CALL HCHAR(13,10,43,8)
280 CALL HCHAR(15,10,58)
290 CALL HCHAR(13,18,44)
```

```
300 CALL HCHAR(13,9,45)
310 CALL HCHAR(14,8,45)
320 CALL HCHAR(14,19,46)
330 CALL HCHAR(14,20,47)
340 CALL HCHAR(15,20,48)
350 CALL HCHAR(15,21,49)
360 CALL HCHAR(14,26,43)
370 CALL HCHAR(14,27,50)
380 CALL HCHAR(15,26,50)
390 CALL HCHAR(14,25,45)
400 CALL HCHAR(13,24,43,2)
410 CALL HCHAR(13,26,51)
420 CALL HCHAR(13,27,52)
430 CALL HCHAR(13,28,43,2)
440 CALL HCHAR(15,25,43)
450 CALL HCHAR(12,24,44)
460 CALL HCHAR(12,29,45)
470 CALL HCHAR(15,24,45)
480 CALL HCHAR(16,1,40,7)
490 CALL HCHAR(16,8,56,18)
500 CALL HCHAR(16,26,40,7)
510 FOR DELAY=1 TO 100
520 NEXT DELAY
530 FOR A=1 TO 2
540 FOR X=12 TO 8 STEP -1
550 CALL HCHAR(X,11,64)
560 NEXT X
570 FOR X=12 TO 8 STEP -1
580 CALL HCHAR(X,11,32)
590 NEXT X
600 NEXT A
610 CALL HCHAR(16,1,41,7)
620 CALL HCHAR(16,8,57,17)
630 CALL HCHAR(16,25,59)
640 CALL HCHAR(16,26,41,7)
650 FOR DELAY=1 TO 100
660 NEXT DELAY
670 GOTO 250
680 END
```

FLY TRAP

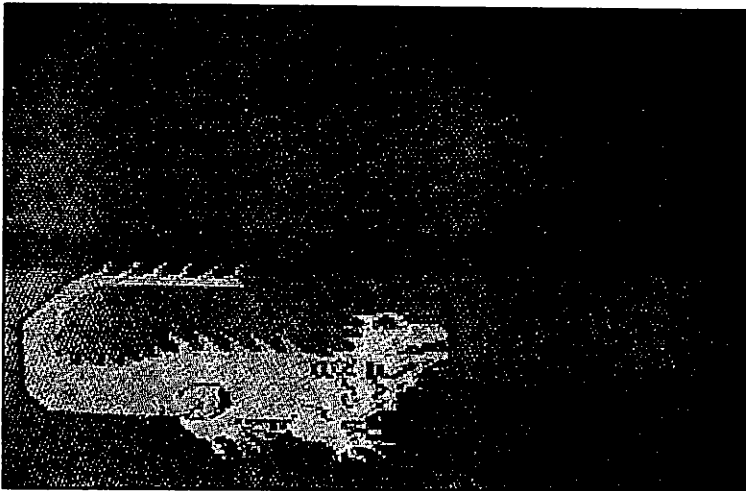
The READ and DATA statements are used in this program to define characters. The GOSUB was used to save some typing in the control of the fly's movement.

PROGRAM STRUCTURE

30-660	define characters
670-720	set colors
730-1340	place lizard on screen
1350-2090	give coordinates for fly
2100-2190	control tongue movement
2200	keeps program running
2210-2280	a subroutine to control fly's movement

RECOMMENDATIONS FOR ALTERATIONS

Put more than one fly on screen at a time. Add more background: place the lizard in its natural habitat with rocks, cacti, etc.



```

10 REM FLY TRAP
20 CALL CLEAR
30 READ A
40 IF A=-1 THEN 670
50 READ A$
60 CALL CHAR(A,A$)
70 GOTO 30
80 DATA 33,000F070303EF0F07
90 DATA 34,3FF3E1C0C0D0FFFF
100 DATA 35,10C0E0F0FFFFFFFF
110 DATA 36,00000000F8FCFEFF
120 DATA 37,037F3F1F0F9FFFFF
130 DATA 38,FFFFFFFFF8E010FF
140 DATA 39,FFFEF0000000FFFF
150 DATA 40,FFFFFFFFFFFFFFFF
160 DATA 41,FFE7EFD0FBFBFCFEF
170 DATA 42,FFFF7F6F6F6F6730
180 DATA 43,FFFFF3E3C79F3E7C
190 DATA 44,FF9820401
200 DATA 45,BF7F3FDFEFF7BFBF
210 DATA 46,BFBFCFF7F7CFDFDF
220 DATA 47,FCDCBCB1F0A08080
230 DATA 48,BFBFDFFFFFFFFFFFF
240 DATA 49,F7EFEFF7E0C01010
250 DATA 50,013E8FE0DFFFEFFF
260 DATA 51,000000F0F00000F0
270 DATA 52,040283C1E1F0F8FF
280 DATA 53,000080C0E0F0F8FF
290 DATA 54,0000000000180F07
300 DATA 55,FFFEAD5B5B5D6EB7
310 DATA 56,00000000030F3FFF
320 DATA 57,183060E0C3FFFFFFFF
330 DATA 58,3060C0C080C0E0F0
340 DATA 59,FFFEFCF8F0E0C080
350 DATA 60,0103070F1F3F7FFF
360 DATA 61,FFFEFCF8F8F8F8F8
370 DATA 62,F8F0F0E0E0C0C0C0
380 DATA 63,C0CCEFEFEFEFFFF
390 DATA 64,FF7F3F1F0F070301
400 DATA 65,0004070301004060
410 DATA 66,FFFFFFFF0EFD0BF7F
420 DATA 67,FFFFFFFF0FF7FBF1F1
430 DATA 68,DFBFBFBFD0EFF7FB
440 DATA 69,FFFFFFFFEFD0FDFD
450 DATA 70,FFFFFCDF9F7EF1F
460 DATA 71,F1F1F1F1FBF7EF9F
470 DATA 72,FFFFF8FFFFFFDFBF
480 DATA 73,FFFFFF3FCFEFEFF7
490 DATA 74,C0FFFF7F3F1F0F07
500 DATA 75,FFFFFFFFFFFFFFFFC3

```

```

510 DATA 76,A092FDE7F8F7F3F8
520 DATA 77,0066EEE67F8080F0
530 DATA 78,7FFFFFFF7F7F3F0F
540 DATA 79,C160707038380808
550 DATA 80,DCCE676331303000
560 DATA 81,3F0F070303030301
570 DATA 82,00002030383CFFFF
580 DATA 83,080C87C7FFFFFFFF
590 DATA 84,7F1FFFFFFFFFFFFFFF
600 DATA 85,FFC0E070180C0000
610 DATA 86,FFFFFFE1F0303E0F01
620 DATA 138,FFFFFFFFF1E010FF
630 DATA 139,FFFEF0000000FFFF
640 DATA 144,000000FFFFFFFF0000
650 DATA 152,0000241818240000
660 DATA -1
670 FOR B=1 TO 7
680 CALL COLOR(B,12,1)
690 NEXT B
700 CALL COLOR(14,12,9)
710 CALL COLOR(15,9,1)
720 CALL SCREEN(6)
730 CALL HCHAR(15,15,33)
740 CALL HCHAR(15,16,34)
750 CALL HCHAR(15,17,35)
760 CALL HCHAR(15,18,36)
770 CALL HCHAR(16,15,37)
780 CALL HCHAR(16,16,40)
790 CALL HCHAR(16,17,38)
800 CALL HCHAR(16,18,39)
810 CALL HCHAR(17,15,41)
820 CALL HCHAR(17,16,42)
830 CALL HCHAR(17,17,43)
840 CALL HCHAR(17,18,44)
850 CALL HCHAR(18,15,45)
860 CALL HCHAR(18,16,46)
870 CALL HCHAR(18,17,47)
880 CALL HCHAR(19,15,48)
890 CALL HCHAR(19,16,49)
900 CALL HCHAR(20,14,40)
910 CALL HCHAR(20,15,50)
920 CALL HCHAR(20,16,51)
930 CALL HCHAR(21,16,85)
940 CALL HCHAR(21,15,86)
950 CALL HCHAR(16,9,52,4)
960 CALL HCHAR(16,13,53)
970 CALL HCHAR(16,14,54)
980 CALL HCHAR(17,9,40,5)
990 CALL HCHAR(18,11,40,4)

```

```
1000 CALL HCHAR(19,12,40)
1010 CALL HCHAR(17,14,55)
1020 CALL HCHAR(13,5,56)
1030 CALL HCHAR(13,6,57,5)
1040 CALL HCHAR(13,11,58)
1050 CALL HCHAR(14,5,59)
1060 CALL HCHAR(14,4,60)
1070 CALL HCHAR(15,3,60)
1080 CALL HCHAR(15,4,61)
1090 CALL HCHAR(16,4,62)
1100 CALL HCHAR(17,4,63)
1110 CALL VCHAR(16,3,40,3)
1120 CALL HCHAR(19,3,64)
1130 CALL HCHAR(18,3,40,6)
1140 CALL HCHAR(19,4,40,4)
1150 CALL HCHAR(16,8,65)
1160 CALL HCHAR(18,9,66)
1170 CALL HCHAR(18,10,67)
1180 CALL HCHAR(19,8,69)
1190 CALL HCHAR(19,9,70)
1200 CALL HCHAR(19,10,71)
1210 CALL HCHAR(19,11,40)
1220 CALL HCHAR(19,13,72)
1230 CALL HCHAR(19,14,73)
1240 CALL HCHAR(20,9,74)
1250 CALL HCHAR(20,10,75)
1260 CALL HCHAR(20,11,76)
1270 CALL HCHAR(20,12,77)
1280 CALL HCHAR(20,13,78)
1290 CALL HCHAR(21,10,79)
1300 CALL HCHAR(21,11,80)
1310 CALL HCHAR(21,14,81)
1320 CALL HCHAR(17,5,82,2)
1330 CALL HCHAR(17,7,83)
1340 CALL HCHAR(17,8,84)
1350 A=10
1360 B=32
1370 GOSUB 2210
1380 A=9
1390 B=31
1400 GOSUB 2210
1410 A=9
1420 B=30
1430 GOSUB 2210
1440 A=9
1450 B=29
1460 GOSUB 2210
1470 A=10
1480 B=28
```


1490 GOSUB 2210
1500 A=9
1510 B=28
1520 GOSUB 2210
1530 A=9
1540 B=27
1550 GOSUB 2210
1560 A=8
1570 B=26
1580 GOSUB 2210
1590 A=7
1600 B=25
1610 GOSUB 2210
1620 A=7
1630 B=24
1640 GOSUB 2210
1650 A=7
1660 B=23
1670 GOSUB 2210
1680 A=7
1690 B=22
1700 GOSUB 2210
1710 A=7
1720 B=21
1730 GOSUB 2210
1740 A=8
1750 B=21
1760 GOSUB 2210
1770 A=9
1780 B=21
1790 GOSUB 2210
1800 A=10
1810 B=22
1820 GOSUB 2210
1830 A=11
1840 B=23
1850 GOSUB 2210
1860 A=12
1870 B=24
1880 GOSUB 2210
1890 A=13
1900 B=24
1910 GOSUB 2210
1920 A=14
1930 B=25
1940 GOSUB 2210
1950 A=14
1960 B=26
1970 GOSUB 2210

```
1980 A=14
1990 B=27
2000 GOSUB 2210
2010 A=14
2020 B=28
2030 GOSUB 2210
2040 A=15
2050 B=27
2060 GOSUB 2210
2070 A=16
2080 B=27
2090 GOSUB 2210
2100 CALL HCHAR(16,17,138)
2110 CALL HCHAR(16,18,139)
2120 CALL HCHAR(16,19,144,9)
2130 FOR X=27 TO 19 STEP -1
2140 CALL HCHAR(16,X,32)
2150 NEXT X
2160 CALL HCHAR(16,18,39)
2170 CALL HCHAR(16,17,38)
2180 FOR DELAY=1 TO 500
2190 NEXT DELAY
2200 GOTO 1440
2210 X=A
2220 Y=B
2230 CALL HCHAR(X,Y,152)
2240 CALL SOUND(200,-3,20)
2250 IF X<>16 THEN 2270
2260 IF Y=27 THEN 2100 ELSE 2270
2270 CALL HCHAR(X,Y,32)
2280 RETURN
2290 END
```

JUST GOOD OL' BOYS

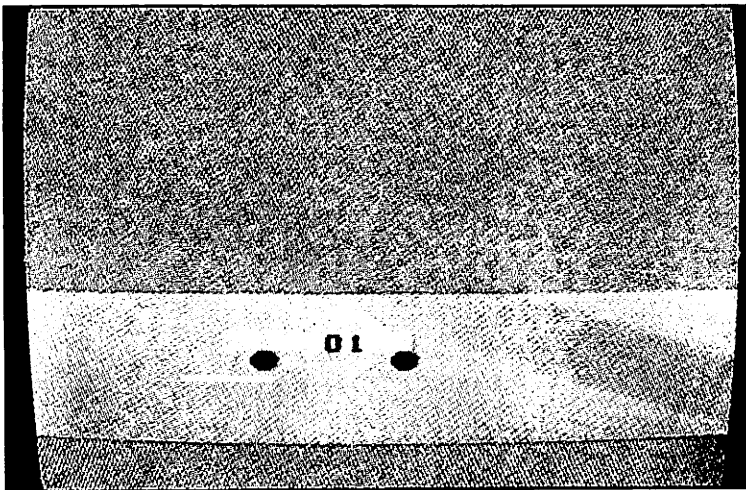
Type in this program and you will find yourself running moonshine on a back road in a little county down south.

PROGRAM STRUCTURE

20-60	define characters
65-90	set colors
95-140	place car on screen
145-170	control road movements
175	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Place another car on the screen. Make it white with a star on the door and a flashing red light on top.



```

5 REM JUST GOOD OL' BOYS
10 CALL CLEAR
15 CALL SCREEN(4)
20 CALL CHAR(112,"000000FFFF000000")
25 CALL CHAR(128,"FFFFFFFFFFFFFFFF")
30 CALL CHAR(136,"FFFF")
35 CALL CHAR(137,"80C0E0F0F8FCFEFF")
40 CALL CHAR(139,"0103060C183060C0")
45 CALL CHAR(153,"FFFFFFFFFFFFFFFF")
50 CALL CHAR(120,"3C7EFFFFFFFF7E3C")
55 CALL CHAR(155,"FFFFFF")
60 CALL CHAR(156,"80C0E0F0F8FCFEFF")
65 CALL COLOR(3,2,10)
70 CALL COLOR(11,16,11)
75 CALL COLOR(13,10,1)
80 CALL COLOR(12,2,11)
85 CALL COLOR(14,10,11)
90 CALL COLOR(16,11,15)
95 CALL HCHAR(15,1,153,256)
100 CALL HCHAR(17,10,128,4)
105 CALL HCHAR(18,11,120)
110 CALL HCHAR(16,13,139)
115 CALL HCHAR(16,14,136,2)
120 CALL VCHAR(17,14,48)
125 CALL VCHAR(17,15,49)
130 CALL HCHAR(16,16,137)
135 CALL HCHAR(17,16,128,2)
140 CALL HCHAR(18,17,120)
145 FOR Y=1 TO 29 STEP 7
150 IF Y=1 THEN 155 ELSE 160
155 CALL HCHAR(19,29,153,4)
160 CALL HCHAR(19,Y,112,4)
165 CALL HCHAR(19,Y,153,4)
170 NEXT Y
175 GOTO 145
180 END

```

PLAY BALL

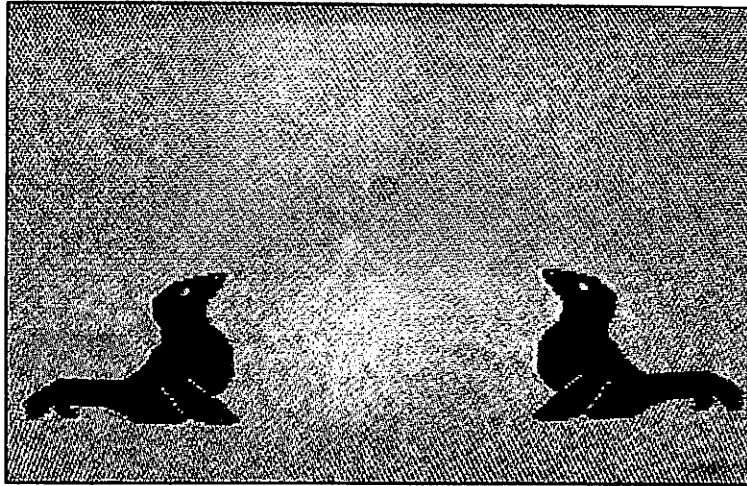
This program uses the READ and DATA statements to define characters. This method may save you some typing time. As you will see, the two seals in the program are having a ball.

PROGRAM STRUCTURE

30-40	set colors
50-620	define characters
630-1210	place seals on screen
1220-1250	give starting coordinates for ball
1260-1470	control sound and ball movement
1480	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Move each seal's head backward as it catches the ball, then forward again as the ball is thrown. Make the ball bigger and add another seal that balances the ball on her nose.



```
10 REM  PLAY BALL
20 CALL CLEAR
30 CALL COLOR(2,9,1)
40 CALL SCREEN(12)
50 READ A
60 IF A=-1 THEN 630
70 READ A$
80 CALL CHAR(A,A$)
90 GOTO 50
100 DATA 40,3C7EFFFFFFFF7E3C
110 DATA 96,3C273F1F1F0F0707
120 DATA 97,00C0F8FFFFE3E1F1
130 DATA 98,0000000080C0E0F0
140 DATA 99,0301010000000000
150 DATA 100,FFFFFFFFFFFFFFFF
160 DATA 101,F8FCFCFCFCFCFCFC
170 DATA 102,01010103070F1F1F
180 DATA 103,F0FCFEFFFFFFFFFFFF
190 DATA 104,FCFCF8F8F0F0F0F0
200 DATA 105,3F3F7F7F7F7F7F7F
210 DATA 106,00000080C0F0F8FC
220 DATA 107,F0F0F8FCFEFEFFFF
230 DATA 108,7F7F7F7F7F7F3F3F
240 DATA 109,FFFFFFFFFFFFBFBF3
250 DATA 110,0F0F1F2B55AAD52A
260 DATA 111,8080C0E0F0F8FCFF
270 DATA 112,1F1F0F0702010307
280 DATA 113,E7CF9F3FFFFFFFFF
290 DATA 114,FFDF1F7E7EFCFCF
```

```

300 DATA 115,FFFFFFFFFDF080
310 DATA 116,9FBF3F7FFFFFFFF7F
320 DATA 117,FFFFFFFFFFFFEF8
330 DATA 118,FFFFFFCF0C0000000
340 DATA 119,FFFF000000000000
350 DATA 120,FFFF54AA54A85000
360 DATA 121,FEFDEA552A150A0A
370 DATA 122,000000000103070F
380 DATA 123,00031FFFFFC7878F
390 DATA 124,3CE4FCF8F8F0E0E0
400 DATA 125,1F3F3F3F3F3F3F3F
410 DATA 126,3F1F1F0F0F0F0F0F
420 DATA 127,808080C0E0F0F8F8
430 DATA 128,0F0F1F3F7F7FFFFF
440 DATA 129,FCFCFEFEFEFEFEFE
450 DATA 130,010103070F1F3FFF
460 DATA 131,FFFFFFFFFDFDFCF
470 DATA 132,FEFEFEFEFEFEFCFC
480 DATA 133,BFBF8FCFC7F7F3FB
490 DATA 134,E7F3F1FCFFFFFFFFF
500 DATA 135,F8F8F0E04000C0E0
510 DATA 136,C08080
520 DATA 137,F0F0F8D4AA55AB54
530 DATA 138,FFFFFF7FFBF0F01
540 DATA 139,F9FDFCFEFFFFFFFFE
550 DATA 140,FFFFFFFFFFFFFF7F
560 DATA 141,FFF3F0F03000000
570 DATA 142,FFFF
580 DATA 143,FFFF28552A150A00
590 DATA 144,7FBF57AA54A850A0
600 DATA 145,00000001030F1F3F
610 DATA 146,1F3F7FFFFFFFFFFFF
620 DATA -1
630 CALL HCHAR(15,9,122)
640 CALL HCHAR(15,10,123)
650 CALL HCHAR(15,11,124)
660 CALL HCHAR(16,9,125)
670 CALL VCHAR(16,10,100,3)
680 CALL HCHAR(17,9,126)
690 CALL HCHAR(17,11,127)
700 CALL HCHAR(18,9,128)
710 CALL HCHAR(18,11,129)
720 CALL HCHAR(19,8,130)
730 CALL HCHAR(19,9,100)
740 CALL HCHAR(19,10,131)
750 CALL HCHAR(19,11,132)
760 CALL HCHAR(20,8,100)
770 CALL HCHAR(20,9,133)
780 CALL HCHAR(20,10,134)

```

```

790 CALL HCHAR(20,11,135)
800 CALL HCHAR(16,11,136)
810 CALL HCHAR(21,11,137)
820 CALL HCHAR(21,10,138)
830 CALL HCHAR(21,9,139)
840 CALL HCHAR(20,6,100,3)
850 CALL HCHAR(20,5,146)
860 CALL HCHAR(20,4,145)
870 CALL HCHAR(21,4,144)
880 CALL HCHAR(21,5,143)
890 CALL HCHAR(21,6,142)
900 CALL HCHAR(21,7,141)
910 CALL HCHAR(21,8,140)
920 CALL HCHAR(15,23,96)
930 CALL HCHAR(15,24,97)
940 CALL HCHAR(15,25,98)
950 CALL HCHAR(16,23,99)
960 CALL HCHAR(16,24,100)
970 CALL HCHAR(16,25,101)
980 CALL HCHAR(17,23,102)
990 CALL HCHAR(17,24,100)
1000 CALL HCHAR(17,25,104)
1010 CALL HCHAR(18,23,105)
1020 CALL HCHAR(18,24,100)
1030 CALL HCHAR(18,25,107)
1040 CALL HCHAR(19,23,108)
1050 CALL HCHAR(19,24,109)
1060 CALL HCHAR(19,25,100)
1070 CALL HCHAR(19,26,111)
1080 CALL HCHAR(20,23,112)
1090 CALL HCHAR(20,24,113)
1100 CALL HCHAR(20,25,114)
1110 CALL HCHAR(20,26,100,3)
1120 CALL HCHAR(20,29,103)
1130 CALL HCHAR(20,30,106)
1140 CALL HCHAR(21,23,110)
1150 CALL HCHAR(21,24,115)
1160 CALL HCHAR(21,25,116)
1170 CALL HCHAR(21,26,117)
1180 CALL HCHAR(21,27,118)
1190 CALL HCHAR(21,28,119)
1200 CALL HCHAR(21,29,120)
1210 CALL HCHAR(21,30,121)
1220 XOLD=14
1230 YOLD=11
1240 XDIR=-1
1250 YDIR=2
1260 FOR A=1 TO 3
1270 XNEW=XOLD+XDIR
1280 YNEW=YOLD+YDIR

```



```
1290 CALL HCHAR(XOLD,YOLD,40)
1300 FOR B=1 TO 15
1310 NEXT B
1320 CALL HCHAR(XOLD,YOLD,32)
1330 XOLD=XNEW
1340 YOLD=YNEW
1350 NEXT A
1360 FOR A=1 TO 3
1370 XNEW=XOLD-XDIR
1380 YNEW=YOLD+YDIR
1390 CALL HCHAR(XOLD,YOLD,40)
1400 FOR B=1 TO 15
1410 NEXT B
1420 CALL HCHAR(XOLD,YOLD,32)
1430 YOLD=YNEW
1440 XOLD=XNEW
1450 NEXT A
1460 CALL SOUND(10,800,0)
1470 YDIR=-YDIR
1480 GOTO 1260
1490 END
```

FRESH EGGS TODAY

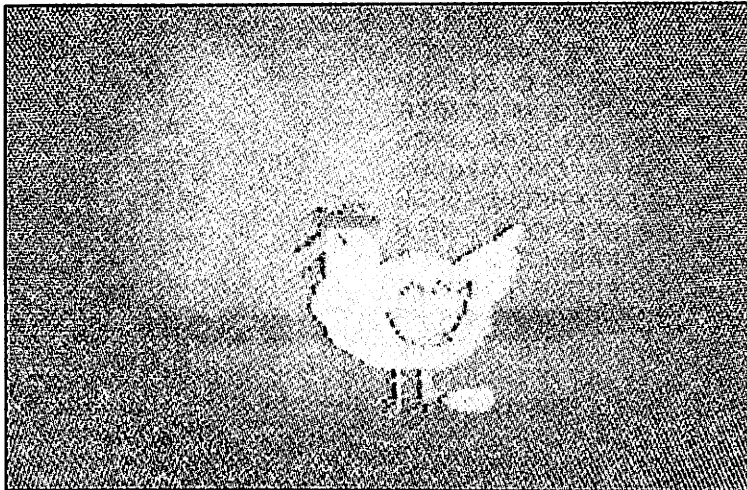
Here I made use of the READ and DATA statements not only in the defining of characters, but also for the CALL HCHAR statements. Movement was created by printing two blank lines.

PROGRAM STRUCTURE

30-100	set colors
110-540	define characters
560-720	place chicken on screen
730-740	delay
750-760	creates movement
770-820	egg is laid
840	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Put in a clucking sound as the egg is laid. Draw more hens and have your own chicken farm. Make all of them lay eggs.



```

10 REM   FRESH EGGS TODAY
20 CALL CLEAR
30 CALL SCREEN(6)
40 CALL COLOR(2,9,1)
50 CALL COLOR(3,11,1)
60 CALL COLOR(4,15,1)
70 CALL COLOR(5,15,1)
80 CALL COLOR(6,15,1)
90 CALL COLOR(7,15,1)
100 CALL COLOR(8,15,1)
110 READ A
120 IF A=-1 THEN 550
130 READ A$
140 CALL CHAR(A,A$)
150 GOTO 110
160 DATA 40,0003070F0F070301
170 DATA 41,0387CFFFFFFFFFFFFF
180 DATA 42,F0E0C0B0FCFEFFFE
190 DATA 44,0103070F1F3F3F3C
200 DATA 48,0103070F1F7FFF
210 DATA 49,1818183C3C181818
220 DATA 50,181E18F818284888
230 DATA 56,C0E0F0F8F8FCFCFE
240 DATA 64,FFE7E7F3F3FFFFFF
250 DATA 65,FFFFFFFFFFFFFFFF
260 DATA 66,FEFEFCFCFFFFFFFF
270 DATA 67,030F3FFFFFFFFFFFFF
280 DATA 68,F0FCFFFFFFFFFFFFFFFF
290 DATA 69,000001C3F7FFFFFF
300 DATA 70,3F7FFFFFFFFFFFFFFFF
310 DATA 71,000000000001071F
320 DATA 72,030F1F3F7EFEFCFC
330 DATA 73,FCFCFEFEFEFCFCFC
340 DATA 74,01030707070F0F0F
350 DATA 75,FFFFFFEFCF9E3FFFF
360 DATA 76,FF7F7E9ECCE1FFFF
370 DATA 77,FFFF3F3FBFBF3F7F
380 DATA 78,F8F0F0F8F0E0C0C0
390 DATA 79,1F1F0F0F07070301
400 DATA 80,FFFFFFBFBFBFD9DF
410 DATA 81,FEFEFEFEFCDF9FB
420 DATA 82,FFFFFFEFEFEFEFE
430 DATA 83,FFFFFFFF7F7F3F1F
440 DATA 84,CFEFE7F7F3F9FCFF
450 DATA 85,FFFFFFFFFFFF3F80
460 DATA 86,F3F7E71F1FCF2F7F
470 DATA 87,FEFEFEFEF8F0F0E0
480 DATA 88,070301
490 DATA 89,FFFFFFFF3F1F07

```

```

500 DATA 90,FFFFFFFFFEF8FOCO
510 DATA 91,E0C0C080
520 DATA 92,071F3F7F7F3F1F07
530 DATA 93,FCFEFEFFFFFFEFEFC
540 DATA -1
550 CALL CLEAR
560 READ X,Y,Z
570 IF X=-1 THEN 730
580 CALL HCHAR(X,Y,Z)
590 GOTO 560
600 DATA 12,15,40,12,16,41,12,17,42
610 DATA 13,15,48,13,16,64,13,17,56,13,21,71,13,22,72
620 DATA 14,15,44,14,16,65,14,17,66,14,18,67,14,19,68,14,20
630 DATA 69,14,21,70,14,22,73
640 DATA 15,15,74,15,16,65,15,17,65,15,18,75,15,19,76,15,20
650 DATA 76,15,21,77,15,22,78
660 DATA 16,15,79,16,16,65,16,17,65,16,18,80,16,19,65,16,20
670 DATA 81,16,21,82
680 DATA 17,16,83,17,17,65,17,18,84,17,19,85,17,20,86,17,21
690 DATA 87
700 DATA 18,16,88,18,17,89,18,18,65,18,19,65,18,20,90,18,21
710 DATA 91
720 DATA -1,-1,-1
730 FOR DELAY=1 TO 1200
740 NEXT DELAY
750 PRINT
760 PRINT
770 CALL HCHAR(17,18,49,2)
780 CALL HCHAR(18,18,50,2)
790 CALL HCHAR(18,20,92)
800 CALL HCHAR(18,21,93)
810 FOR DELAY=1 TO 3000
820 NEXT DELAY
830 RESTORE 560
840 GOTO 550
850 END

```

BIRD WATCHING

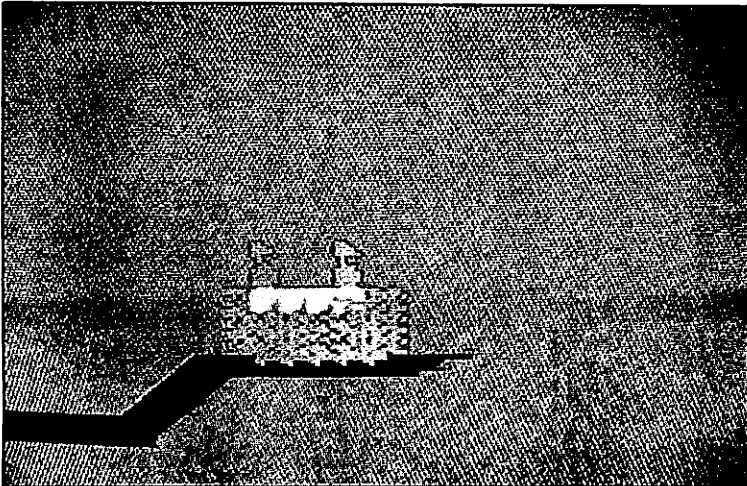
Through time-lapse computer software, you can now see the hatching of a rare breed of bird found only in Silicon Valley.

PROGRAM STRUCTURE

30-100	set colors
110-390	define characters
400-670	place nest on screen
700-730	place cracks in eggs
760-810	place birds on screen
820-860	control sound
870	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Add leaves to the branch or have the birds' heads move.



```

10 REM    BIRD WATCHING
20 CALL CLEAR
30 CALL COLOR(2,2,6)
40 CALL COLOR(3,16,6)
50 CALL COLOR(4,16,11)
60 CALL COLOR(5,11,6)
70 CALL COLOR(6,16,1)
80 CALL COLOR(7,2,11)
90 CALL COLOR(8,10,1)
100 CALL SCREEN(6)
110 READ A
120 IF A=-1 THEN 420
130 READ A$
140 CALL CHAR(A,A$)
150 GOTO 110
160 DATA 80,00000E8EEEEFFFFFFF
170 DATA 40,FFFFFFFFFFFFFFFFFFFF
180 DATA 41,FFFFFFFF
190 DATA 42,0103070F1F3F7FFF
200 DATA 43,FFFEFCF8FOE0C080
210 DATA 44,FFFFFFEFCF8FF0
220 DATA 45,FFFF
230 DATA 56,FFFFFFFFEFCFCFCF0
240 DATA 57,FF7F772300D007E0
250 DATA 49,000F1F3F7F7F7F3F
260 DATA 50,00F0F8FCFEFEFEFE
270 DATA 51,03070F1F3F7F7F7F
280 DATA 52,F0F8FCFCFCF8FFFF
290 DATA 64,FEFF37D7EBDF3FFF
300 DATA 65,E0F0D8E8E0E0F878
310 DATA 66,F5DFDEB7FDFBFEBF
320 DATA 72,000F1F2B757D7E3F
330 DATA 73,00F0F7FCEE9E7EFE
340 DATA 74,03070F1E3D7D7D77
350 DATA 75,F0E86CFCFCF8FFFF
360 DATA 88,70F8FCFE73202008
370 DATA 89,3C7EFFFFFFFFF7E3C
380 DATA 90,60F8FCFE7F202020
390 DATA -1
400 CALL VCHAR(12,12,32,2)
410 CALL VCHAR(12,15,32,2)
420 CALL HCHAR(20,1,40,7)
430 CALL HCHAR(21,1,41,7)
440 CALL HCHAR(20,8,43)
450 CALL HCHAR(19,7,42)
460 CALL HCHAR(19,8,40)
470 CALL HCHAR(19,9,43)
480 CALL HCHAR(18,8,42)
490 CALL HCHAR(18,9,40)

```

```

500 CALL HCHAR(18,10,43)
510 CALL HCHAR(17,9,42)
520 CALL HCHAR(17,10,40,2)
530 CALL HCHAR(17,12,80,5)
540 CALL HCHAR(17,17,40)
550 CALL HCHAR(17,18,44)
560 CALL HCHAR(17,19,45)
570 CALL HCHAR(13,12,49)
580 CALL HCHAR(13,13,50)
590 CALL HCHAR(13,14,51)
600 CALL HCHAR(13,15,52)
610 CALL HCHAR(14,12,56,3)
620 CALL HCHAR(14,15,57)
630 CALL HCHAR(14,16,66)
640 CALL VCHAR(14,11,64,3)
650 CALL VCHAR(14,17,65,3)
660 CALL HCHAR(15,12,66,5)
670 CALL HCHAR(16,12,66,5)
680 FOR DELAY=1 TO 1600
690 NEXT DELAY
700 CALL HCHAR(13,12,72)
710 CALL HCHAR(13,13,73)
720 CALL HCHAR(13,14,74)
730 CALL HCHAR(13,15,75)
740 FOR DELAY=1 TO 900
750 NEXT DELAY
760 CALL HCHAR(12,12,88)
770 CALL HCHAR(13,12,89)
780 CALL HCHAR(13,13,32)
790 CALL HCHAR(12,15,90)
800 CALL HCHAR(13,14,32)
810 CALL HCHAR(13,15,89)
820 FOR A=1 TO 17
830 CALL SOUND(10,1500,0,-7,5)
840 FOR DELAY=1 TO 100
850 NEXT DELAY
860 NEXT A
870 GOTO 400
880 END

```

FACE TO FACE

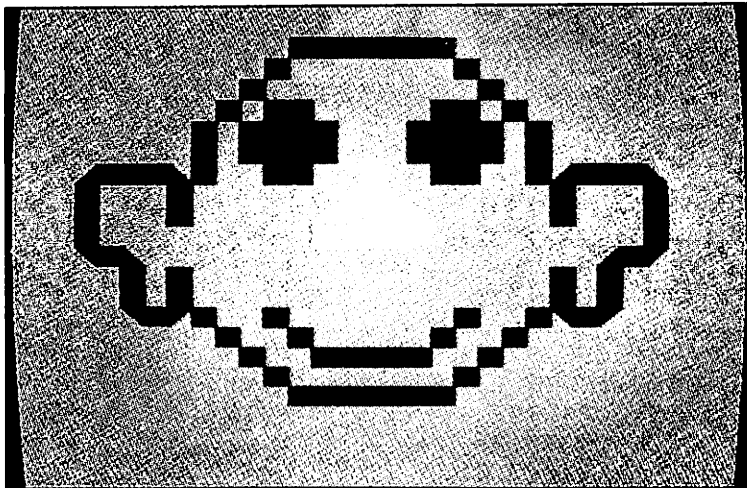
Did you ever want to make a face at someone, but felt stupid doing so? Although this program is lengthy, it lets you make all the faces you want. By using the INPUT statement, you may interact with the program to create many faces.

PROGRAM STRUCTURE

8-12	state instructions
16-24	define characters for face outline
30-118	place face outline on screen
120-136	determine which set of facial characters are requested
138-428	define facial characters and place them on screen
430-432	time delay between faces
434	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Make up your own faces to put into the program.




```

2 REM FACE TO FACE
4 CALL COLOR(3,2,1)
6 CALL CLEAR
8 INPUT "EYES - ENTER 1,2, OR 3 ":A
10 INPUT "NOSE - ENTER 1,2, OR 3 ":B
12 INPUT "MOUTH- ENTER 1,2, OR 3 ":C
14 CALL CLEAR
16 CALL CHAR(97,"FFFFFFFFFFFFFFFF")
18 CALL CHAR(98,"FFFEFCF8F0E0C080")
20 CALL CHAR(99,"FF7F3F1F0F070301")
22 CALL CHAR(100,"0103070F1F3F7FFF")
24 CALL CHAR(101,"80C0E0F0F8FCFEFF")
26 CALL COLOR(9,2,1)
28 CALL SCREEN(10)
30 CALL HCHAR(3,13,97,7)
32 CALL HCHAR(4,12,97)
34 CALL HCHAR(4,20,97)
36 CALL HCHAR(5,11,97)
38 CALL HCHAR(5,21,97)
40 CALL HCHAR(6,10,97)
42 CALL HCHAR(6,22,97)
44 CALL VCHAR(7,9,97,3)
46 CALL VCHAR(7,23,97,3)
48 CALL VCHAR(9,8,101)
50 CALL VCHAR(9,24,100)
52 CALL VCHAR(10,8,97,2)
54 CALL VCHAR(10,24,97,2)
56 CALL HCHAR(9,5,97,3)
58 CALL HCHAR(9,25,97,3)
60 CALL HCHAR(9,4,100)
62 CALL HCHAR(9,28,101)
64 CALL VCHAR(10,4,97,3)
66 CALL VCHAR(10,28,97,3)
68 CALL VCHAR(13,4,99)
70 CALL VCHAR(13,28,98)
72 CALL VCHAR(13,5,97)
74 CALL VCHAR(13,27,97)
76 CALL VCHAR(13,6,101)
78 CALL VCHAR(13,26,100)
80 CALL VCHAR(14,6,97,2)
82 CALL VCHAR(14,26,97,2)
84 CALL VCHAR(16,6,99)
86 CALL VCHAR(16,26,98)
88 CALL VCHAR(16,7,97)
90 CALL VCHAR(16,25,97)
92 CALL VCHAR(16,8,98)
94 CALL VCHAR(16,24,99)
96 CALL VCHAR(14,8,97,2)
98 CALL VCHAR(14,24,97,2)

```

```

100 CALL VCHAR(16,9,97)
102 CALL VCHAR(16,23,97)
104 CALL VCHAR(17,10,97)
106 CALL HCHAR(17,22,97)
108 CALL HCHAR(18,11,97)
110 CALL HCHAR(18,21,97)
112 CALL HCHAR(19,12,97)
114 CALL HCHAR(19,20,97)
116 CALL HCHAR(20,13,97,7)
118 CALL HCHAR(23,1,32,31)
120 IF A=1 THEN 138
122 IF A=2 THEN 176
124 IF A=3 THEN 198
126 IF B=1 THEN 244
128 IF B=2 THEN 286
130 IF B=3 THEN 314
132 IF C=1 THEN 338
134 IF C=2 THEN 364
136 IF C=3 THEN 380
138 CALL COLOR(11,6,1)
140 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
142 CALL CHAR(113,"FFFEFCFBFOEOCOBO")
144 CALL CHAR(114,"FF7F3F1F0F070301")
146 CALL CHAR(115,"0103070F1F3F7FFF")
148 CALL CHAR(116,"80COEOFOFBFCFEFF")
150 CALL VCHAR(7,11,115)
152 CALL VCHAR(7,12,116)
154 CALL VCHAR(8,11,112,2)
156 CALL VCHAR(8,12,112,2)
158 CALL VCHAR(10,11,114)
160 CALL VCHAR(10,12,113)
162 CALL VCHAR(7,20,115)
164 CALL VCHAR(7,21,116)
166 CALL VCHAR(8,20,112,2)
168 CALL VCHAR(8,21,112,2)
170 CALL VCHAR(10,20,114)
172 CALL VCHAR(10,21,113)
174 GOTO 126
176 CALL COLOR(10,2,1)
178 CALL CHAR(104,"FFFFFFFFFFFFFFFF")
180 CALL HCHAR(6,12,104,2)
182 CALL HCHAR(7,11,104,4)
184 CALL HCHAR(8,11,104,4)
186 CALL HCHAR(9,12,104,2)
188 CALL HCHAR(6,19,104,2)
190 CALL HCHAR(7,18,104,4)
192 CALL HCHAR(8,18,104,4)
194 CALL HCHAR(9,19,104,2)
196 GOTO 126
198 CALL COLOR(12,16,1)

```

```

200 CALL COLOR(13,6,16)
202 CALL CHAR(128,"FFFFFFFFFFFFFFFF")
204 CALL CHAR(121,"FFFFFFFFFFFFFFFF")
206 CALL CHAR(122,"0103070F1F3F7FFF")
208 CALL CHAR(123,"FF7F3F1F0F070301")
210 CALL CHAR(124,"80C0E0F0F8FCFEFF")
212 CALL CHAR(125,"FFFEFCF8F0E0C080")
214 CALL VCHAR(7,11,122)
216 CALL VCHAR(8,11,123)
218 CALL VCHAR(7,12,121,2)
220 CALL VCHAR(7,13,128,2)
222 CALL VCHAR(7,14,121,2)
224 CALL VCHAR(7,15,124)
226 CALL VCHAR(8,15,125)
228 CALL VCHAR(7,17,122)
230 CALL VCHAR(8,17,123)
232 CALL VCHAR(7,18,121,2)
234 CALL VCHAR(7,19,128,2)
236 CALL VCHAR(7,20,121,2)
238 CALL VCHAR(7,21,124)
240 CALL VCHAR(8,21,125)
242 GOTO 126
244 CALL COLOR(1,4,1)
246 CALL CHAR(37,"FFFFFFFFFFFFFFFF")
248 CALL CHAR(33,"FFFEFCF8F0E0C080")
250 CALL CHAR(34,"FF7F3F1F0F070301")
252 CALL CHAR(35,"0103070F1F3F7FFF")
254 CALL CHAR(36,"80C0E0F0F8FCFEFF")
256 CALL HCHAR(9,15,37,3)
258 CALL HCHAR(10,14,37,5)
260 CALL HCHAR(11,13,37,7)
262 CALL HCHAR(12,13,37,7)
264 CALL HCHAR(13,14,37,5)
266 CALL HCHAR(14,15,37,3)
268 CALL HCHAR(9,14,35)
270 CALL HCHAR(10,13,35)
272 CALL HCHAR(13,13,34)
274 CALL HCHAR(14,14,34)
276 CALL HCHAR(14,18,33)
278 CALL HCHAR(13,19,33)
280 CALL HCHAR(10,19,36)
282 CALL HCHAR(9,18,36)
284 GOTO 132
286 CALL COLOR(2,11,1)
288 CALL CHAR(40,"0103070F1F3F7FFF")
290 CALL CHAR(41,"80C0E0F0F8FCFEFF")
292 CALL CHAR(42,"FFFFFFFFFFFFFFFF")
294 CALL HCHAR(9,15,40)
296 CALL HCHAR(9,17,41)

```

```

298 CALL HCHAR(9,16,42)
300 CALL HCHAR(10,15,42,3)
302 CALL HCHAR(11,15,42,3)
304 CALL HCHAR(12,14,42,5)
306 CALL HCHAR(13,13,42,2)
308 CALL HCHAR(13,16,42)
310 CALL HCHAR(13,18,42,2)
312 GOTO 132)
314 CALL COLOR(3,15,1)
316 CALL CHAR(55,"FFFFFFFFFFFFFFFF")
318 CALL CHAR(52,"0103070F1F3F7FEFFF")
320 CALL CHAR(53,"80C0E0F0F8FCFEFF")
322 CALL VCHAR(9,16,55,2)
324 CALL VCHAR(10,15,52)
326 CALL VCHAR(10,17,53)
328 CALL HCHAR(11,14,52)
330 CALL HCHAR(11,15,55,3)
332 CALL HCHAR(11,18,53)
334 CALL HCHAR(12,14,55,5)
336 GOTO 132
338 CALL COLOR(4,14,1)
340 CALL CHAR(56,"FFFFFFFFFFFFFFFF")
342 CALL CHAR(57,"0103070F1F3F7FFF")
344 CALL CHAR(58,"80C0E0F0F8FCFEFF")
346 CALL CHAR(59,"FF7F3F1F0F070301")
348 CALL CHAR(60,"FFFEFCF8F0E0C080")
350 CALL HCHAR(16,13,57)
352 CALL HCHAR(16,14,56,5)
354 CALL HCHAR(16,19,58)
356 CALL HCHAR(17,13,59)
358 CALL HCHAR(17,14,56,5)
360 CALL HCHAR(17,19,60)
362 GOTO 430
364 CALL COLOR(5,2,1)
366 CALL CHAR(64,"FFFFFFFFFFFFFFFF")
368 CALL HCHAR(16,12,64)
370 CALL HCHAR(17,13,64)
372 CALL HCHAR(18,14,64,5)
374 CALL HCHAR(17,19,64)
376 CALL HCHAR(16,20,64)
378 GOTO 430
380 CALL COLOR(6,2,1)
382 CALL COLOR(4,7,1)
384 CALL CHAR(76,"FF7F3F1F0F070301")
386 CALL CHAR(75,"FFFFFFFFFFFFFFFF")
388 CALL CHAR(74,"FFFEFCF8F0E0C080")
390 CALL CHAR(56,"FFFFFFFFFFFFFFFF")
392 CALL CHAR(60,"FFFEFCF8F0E0C080")
394 CALL CHAR(59,"FF7F3F1F0F070301")

```

```
396 CALL HCHAR(16,12,75,10)
398 CALL HCHAR(17,12,76)
400 CALL HCHAR(17,13,75)
402 CALL HCHAR(18,13,76)
404 CALL HCHAR(18,14,75)
406 CALL HCHAR(17,20,75)
408 CALL HCHAR(18,19,75)
410 CALL HCHAR(18,20,74)
412 CALL HCHAR(17,15,56,4)
414 CALL HCHAR(17,21,74)
416 CALL HCHAR(18,15,56,4)
418 CALL HCHAR(19,16,56,2)
420 CALL HCHAR(19,15,59)
422 CALL HCHAR(19,18,60)
424 CALL HCHAR(20,16,59)
426 CALL HCHAR(20,17,60)
428 GOTO 430
430 FOR DELAY=1 TO 2000
432 NEXT DELAY
434 GOTO 4
436 END
```

SNAKE EYES

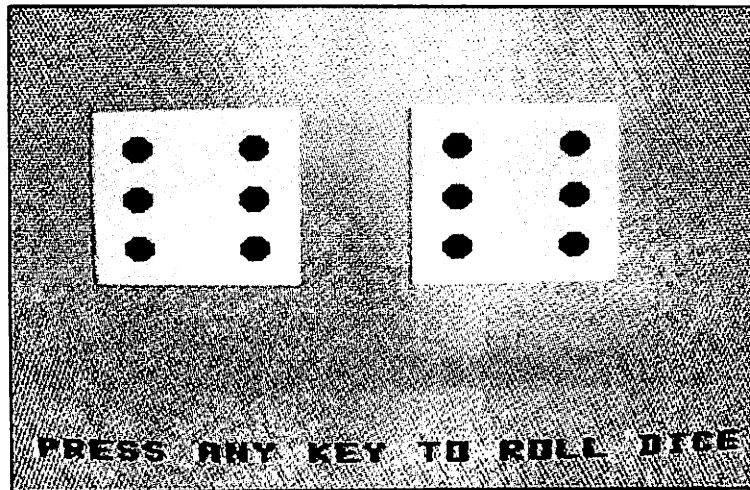
This program appeals to the gambler in all of us: a chance to bet it all on a roll of the dice. It's also a good example of the RND and RANDOMIZE statements.

PROGRAM STRUCTURE

30-40	set colors
50-60	define characters
80-210	place blank dice on screen
220-230	wait for a key to be pressed
240-540	place first die on screen
550-820	place second die on screen
850	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

Use the INPUT statement to place bets, then keep track of the winnings.



```

10 REM SNAKE EYES
20 CALL CLEAR
30 CALL COLOR(2,2,16)
40 CALL SCREEN(8)
50 CALL CHAR(40,"00")
60 CALL CHAR(41,"3C7EFFFFFFFF7E3C")
70 PRINT " PRESS ANY KEY TO ROLL DICE"
80 CALL VCHAR(9,8,40,7)
90 CALL VCHAR(9,9,40,7)
100 CALL VCHAR(9,10,40,7)
110 CALL VCHAR(9,11,40,7)
120 CALL VCHAR(9,12,40,7)
130 CALL VCHAR(9,13,40,7)
140 CALL VCHAR(9,14,40,7)
150 CALL VCHAR(9,19,40,7)
160 CALL VCHAR(9,20,40,7)
170 CALL VCHAR(9,21,40,7)
180 CALL VCHAR(9,22,40,7)
190 CALL VCHAR(9,23,40,7)
200 CALL VCHAR(9,24,40,7)
210 CALL VCHAR(9,25,40,7)
220 CALL KEY(0,S,STATUS)
230 IF STATUS=0 THEN 220
240 RANDOMIZE
250 A=INT(RND*6+1)
260 ON A GOTO 280,300,330,370,420,480
270 GOTO 220
280 CALL HCHAR(12,11,41)
290 GOTO 550
300 CALL HCHAR(10,9,41)
310 CALL HCHAR(14,13,41)
320 GOTO 550
330 CALL HCHAR(10,9,41)
340 CALL HCHAR(12,11,41)
350 CALL HCHAR(14,13,41)
360 GOTO 550
370 CALL HCHAR(10,9,41)
380 CALL HCHAR(10,13,41)
390 CALL HCHAR(14,9,41)
400 CALL HCHAR(14,13,41)
410 GOTO 550
420 CALL HCHAR(10,9,41)
430 CALL HCHAR(10,13,41)
440 CALL HCHAR(12,11,41)
450 CALL HCHAR(14,9,41)
460 CALL HCHAR(14,13,41)
470 GOTO 550
480 CALL HCHAR(10,9,41)
490 CALL HCHAR(12,9,41)

```

```
500 CALL HCHAR(14,9,41)
510 CALL HCHAR(10,13,41)
520 CALL HCHAR(12,13,41)
530 CALL HCHAR(14,13,41)
540 GOTO 550
550 B=INT(RND*6+1)
560 ON B GOTO 570,590,620,660,710,770
570 CALL HCHAR(12,22,41)
580 GOTO 830
590 CALL HCHAR(10,20,41)
600 CALL HCHAR(14,24,41)
610 GOTO 830
620 CALL HCHAR(10,20,41)
630 CALL HCHAR(12,22,41)
640 CALL HCHAR(14,24,41)
650 GOTO 830
660 CALL HCHAR(10,20,41)
670 CALL HCHAR(10,24,41)
680 CALL HCHAR(14,20,41)
690 CALL HCHAR(14,24,41)
700 GOTO 830
710 CALL HCHAR(10,20,41)
720 CALL HCHAR(10,24,41)
730 CALL HCHAR(12,22,41)
740 CALL HCHAR(14,20,41)
750 CALL HCHAR(14,24,41)
760 GOTO 830
770 CALL HCHAR(10,20,41)
780 CALL HCHAR(10,24,41)
790 CALL HCHAR(12,20,41)
800 CALL HCHAR(12,24,41)
810 CALL HCHAR(14,20,41)
820 CALL HCHAR(14,24,41)
830 FOR DELAY=1 TO 800
840 NEXT DELAY
850 GOTO 80
860 END
```


JUST ONE BAD APPLE

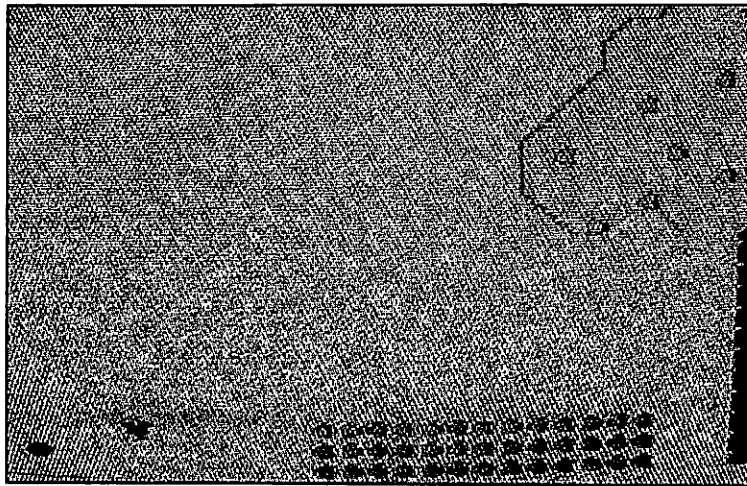
This is a game in which you control movements by pressing a preset key. Instructions are contained in the PRINT statements. You should take care to type the PRINT statements exactly as shown for good screen appearance. See if you can eat all but one apple. Good luck!

PROGRAM STRUCTURE

15-25, 630-705	give directions, if needed
35-65	set colors
70-115	define characters
120-315	place game on screen
320-335	randomly place worm in an apple
340-345	give starting coordinates for bird
350-365	halt program until key is pressed
370-465	control movements of bird
470-475	check to see if worm was found
485-550	control worm movements
585-625	set limits on bird's movements

RECOMMENDATIONS FOR ALTERATIONS

Add a score counter for the apples eaten. Give ten points for each apple, then display the final score at the end of the game.



```
5 REM JUST ONE BAD APPLE
10 CALL CLEAR
15 INPUT " DO YOU NEED INSTRUCTIONS TO PLAY? ENTER Y FOR
YES OR N FOR NO. ":S$
20 IF S$="Y" THEN 25 ELSE 30
25 GOSUB 630
30 CALL CLEAR
35 CALL SCREEN(6)
40 CALL COLOR(2,4,2)
45 CALL COLOR(3,4,1)
50 CALL COLOR(9,9,4)
55 CALL COLOR(10,2,4)
60 CALL COLOR(11,2,4)
65 CALL COLOR(12,2,4)
70 CALL CHAR(40,"0041008200004008")
75 CALL CHAR(41,"FFFEFCF8F0E0C080")
80 CALL CHAR(50,"FFFFFFFFFFFFFFFF")
85 CALL CHAR(51,"0103070F1F3F7FFF")
90 CALL CHAR(52,"FF7F3F1F0F070301")
95 CALL CHAR(53,"FFFEFCF8F0E0C080")
100 CALL CHAR(96,"00183C7E7E7E3C00")
105 CALL CHAR(104,"00001029C9C9121C")
110 CALL CHAR(112,"0006C77C3C7C0804")
115 CALL CHAR(120,"0000003C7EFF7E3C")
120 CALL HCHAR(3,28,51)
125 CALL HCHAR(3,29,50,4)
130 CALL HCHAR(4,27,51)
135 CALL HCHAR(4,28,50,5)
140 CALL HCHAR(5,25,51)
```

```

145 CALL HCHAR(5,26,50,7)
150 CALL HCHAR(6,25,50,8)
155 CALL HCHAR(7,24,51)
160 CALL HCHAR(7,25,50,8)
165 CALL HCHAR(8,23,51)
170 CALL HCHAR(8,24,50,9)
175 CALL HCHAR(9,22,51)
180 CALL HCHAR(9,23,50,10)
185 CALL HCHAR(10,22,50,11)
190 CALL HCHAR(11,22,50,11)
195 CALL HCHAR(12,22,52)
200 CALL HCHAR(12,23,50,10)
205 CALL HCHAR(13,23,52)
210 CALL HCHAR(13,25,53)
215 CALL HCHAR(13,27,52)
220 CALL HCHAR(13,28,50,2)
225 CALL VCHAR(13,30,41)
230 CALL VCHAR(14,30,40,11)
235 CALL VCHAR(13,31,40,12)
240 CALL VCHAR(13,32,40,12)
245 CALL HCHAR(7,29,96)
250 CALL HCHAR(8,26,96)
255 CALL HCHAR(10,23,96)
260 CALL HCHAR(10,27,96)
265 CALL HCHAR(11,29,96)
270 CALL HCHAR(12,26,96)
275 CALL HCHAR(13,24,96)
280 CALL HCHAR(21,1,50,29)
285 CALL HCHAR(22,1,50,29)
290 CALL HCHAR(23,1,50,29)
295 CALL HCHAR(24,1,50,29)
300 CALL HCHAR(22,14,96,13)
305 CALL HCHAR(23,14,96,13)
310 CALL HCHAR(24,14,96,13)
315 CALL HCHAR(23,3,120)
320 RANDOMIZE
325 A=INT((24-22+1)*RND)+22
330 B=INT((26-14+1)*RND)+14
335 CALL HCHAR(A,B,96)
340 XOLD=23
345 YOLD=4
350 CALL KEY(3,KEY,STATUS)
355 IF STATUS=0 THEN 350
360 Z=KEY-64
365 ON Z GOTO 420,445,370,395
370 XNEW=XOLD+1
375 GOSUB 585
380 CALL HCHAR(XOLD,YOLD,50)
385 XOLD=XNEW

```

```

390 GOTO 465
395 XNEW=XOLD-1
400 GOSUB 585
405 CALL HCHAR(XOLD,YOLD,50)
410 XOLD=XNEW
415 GOTO 465
420 YNEW=YOLD+1
425 GOSUB 585
430 CALL HCHAR(XOLD,YOLD,50)
435 YOLD=YNEW
440 GOTO 465
445 YNEW=YOLD-1
450 GOSUB 585
455 CALL HCHAR(XOLD,YOLD,50)
460 YOLD=YNEW
465 CALL HCHAR(XNEW,YNEW,112)
470 IF XNEW=A THEN 475 ELSE 480
475 IF YNEW=B THEN 485
480 GOTO 350
485 CALL HCHAR(A,B,104)
490 FOR I=1 TO 100
495 NEXT I
500 BNEW=B-1
505 IF BNEW=4 THEN 510 ELSE 565
510 CALL HCHAR(A,B,50)
515 A=23
520 CALL HCHAR(A,BNEW,104)
525 FOR T=1 TO 100
530 NEXT T
535 CALL HCHAR(A,BNEW,50)
540 CALL HCHAR(23,3,120)
545 FOR G=1 TO 1500
550 NEXT G
555 CALL CLEAR
560 END
565 CALL HCHAR(A,B,50)
570 B=BNEW
575 CALL HCHAR(A,BNEW,104)
580 GOTO 485
585 IF XNEW>24 THEN 590 ELSE 595
590 XNEW=24
595 IF XNEW<21 THEN 600 ELSE 605
600 XNEW=21
605 IF YNEW<4 THEN 610 ELSE 615
610 YNEW=4
615 IF YNEW>27 THEN 620 ELSE 625
620 YNEW=27
625 RETURN
630 CALL CLEAR

```

```
635 PRINT "YOU ARE A HUNGRY BIRD LOOK- ING FOR WORMS."
640 PRINT "TO FIND WORMS YOU MUST FIRST EAT THE APPLES IN
WHICH THEY HIDE."
641 PRINT "DO THIS BY PASSING OVER THEM."
645 PRINT "ONE OF THE APPLES HAS A WORM WHICH WILL EAT YOU
IF GIVEN THE CHANCE."
650 PRINT "SEE HOW MANY APPLES YOU CAN EAT BEFORE YOU EAT
JUST ONE TOO MANY."
655 PRINT
660 PRINT "PRESS: A TO MOVE RIGHT"
665 PRINT "      B TO MOVE LEFT"
670 PRINT "      C TO MOVE DOWN"
675 PRINT "      D TO MOVE UP"
680 PRINT
685 PRINT "ONCE THE PICTURE APPEARS----PRESS C TO START."
690 PRINT
695 INPUT "ARE YOU READY? (ENTER Y FOR YES) ":A$
700 IF A$="Y" THEN 705
705 RETURN
710 END
```

SKULDUGGERY

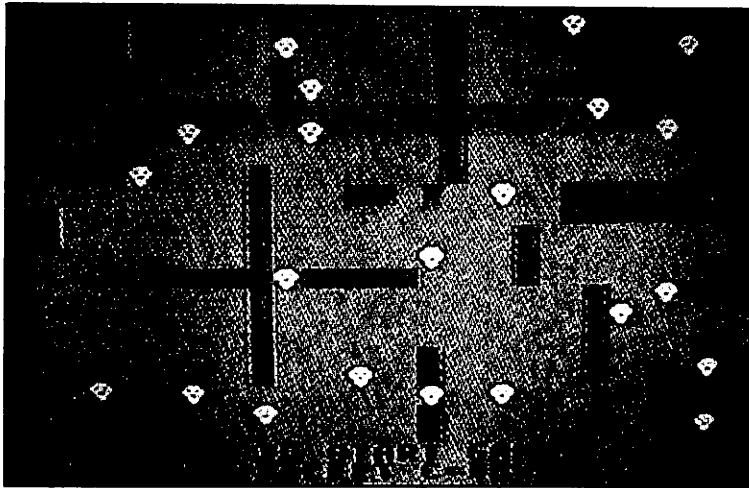
This is a very simple game that uses CALL KEY for its movement. In the game, you are a pirate looking for some treasure (which is in the lower right corner). To get it, you must go through a maze. A skull appears for every step you take. You win if you reach the chest without touching a skull.

PROGRAM STRUCTURE

15-25	set colors
30-45	define characters
50-135	place maze on screen
140-145	place playing instructions on screen
160-165	give coordinates for pirate and chest
170-180	randomly place a skull on screen
185-210	halt program until a key is pressed
215-310	control direction of pirate
315-320	place pirate in new position and keep program running until game is over
325-400	subroutine that checks screen boundaries and checks if game is over

RECOMMENDATIONS FOR ALTERATIONS

In playing the game, you can go through the skulls and the walls. The game was left in this manner to keep the program short. Honesty on your part is required to play the game correctly. It is possible to make the maze harder by increasing the number of walls.



```
5 REM SKULDUGGERY
10 CALL CLEAR
15 CALL SCREEN(6)
20 CALL COLOR(9,16,1)
25 CALL COLOR(13,9,1)
30 CALL CHAR(42,"995A3C3C3C3C2424")
35 CALL CHAR(97,"3C7EDBFF663C3C18")
40 CALL CHAR(128,"FFFFFFFFFFFFFFFF")
45 CALL CHAR(41,"003C7E7EFFFFFFFF")
50 CALL VCHAR(1,5,128,4)
55 CALL VCHAR(3,12,128,5)
60 CALL VCHAR(3,19,128,8)
65 CALL VCHAR(10,11,128,11)
70 CALL VCHAR(13,22,128,3)
75 CALL VCHAR(16,25,128,5)
80 CALL VCHAR(19,18,128,6)
85 CALL VCHAR(20,6,128,5)
90 CALL VCHAR(20,28,128,4)
95 CALL HCHAR(7,6,128,5)
100 CALL HCHAR(6,23,128,6)
105 CALL HCHAR(11,15,128,2)
110 CALL HCHAR(15,5,128,13)
115 CALL HCHAR(9,2,128,5)
120 CALL HCHAR(10,2,128,5)
125 CALL HCHAR(16,28,128,3)
130 CALL HCHAR(11,24,128,11)
135 CALL HCHAR(12,24,128,11)
140 PRINT "PRESS C TO START GAME."
145 PRINT "A=RIGHT B=LEFT C=DOWN D=UP"
```

```

150 CALL CHAR(42,"995A3C3C3C3C2424")
155 CALL HCHAR(21,30,41)
160 XOLD=3
165 YOLD=3
170 RANDOMIZE
175 ROW=INT(21*RND)+1
180 COLUMN=INT(31*RND)+1
185 CALL HCHAR(ROW,COLUMN,97)
190 CALL HCHAR(YOLD,XOLD,42)
195 CALL KEY(3,KEY,STATUS)
200 IF STATUS=0 THEN 195
205 Z=KEY-64
210 ON Z GOTO 215,240,265,290
215 XNEW=XOLD+1
220 GOSUB 330
225 CALL HCHAR(YOLD,XOLD,32)
230 XOLD=XNEW
235 GOTO 315
240 XNEW=XOLD-1
245 GOSUB 330
250 CALL HCHAR(YOLD,XOLD,32)
255 XOLD=XNEW
260 GOTO 315
265 YNEW=YOLD+1
270 GOSUB 330
275 CALL HCHAR(YOLD,XOLD,32)
280 YOLD=YNEW
285 GOTO 315
290 YNEW=YOLD-1
295 GOSUB 330
300 CALL HCHAR(YOLD,XOLD,32)
305 YOLD=YNEW
310 GOTO 315
315 CALL HCHAR(YNEW,XNEW,42)
320 CALL HCHAR(YNEW,XNEW,42)
325 GOTO 175
330 IF XNEW>31 THEN 335 ELSE 340
335 XNEW=31
340 IF XNEW<1 THEN 345 ELSE 350
345 XNEW=1
350 IF YNEW>21 THEN 355 ELSE 360
355 YNEW=21
360 IF YNEW<1 THEN 365 ELSE 370
365 YNEW=1
370 CALL HCHAR(YOLD,XOLD,32)
375 IF XNEW=30 THEN 380 ELSE 410
380 IF YNEW=21 THEN 385 ELSE 410
385 CALL SOUND(1000,220,2)

```



```
390 CALL SOUND (2000,440,2)
395 CALL SOUND (1000,220,2)
400 CALL SOUND (2000,440,2)
405 END
410 RETURN
```

U TURN

This is another interactive program in which you can draw right on the screen. There are nine keys to work with. They are:

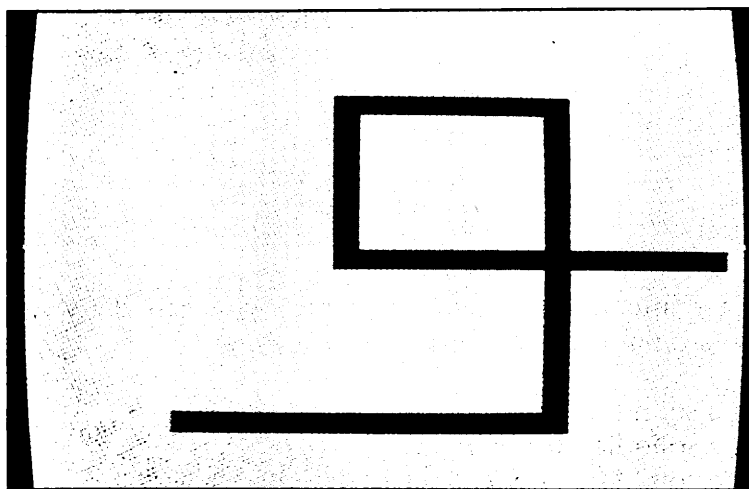
- E draws a solid line up
- D draws a solid line to the right
- X draws a solid line down
- S draws a solid line to the left
- I leaves a blank (or erases) moving upward
- J leaves a blank (or erases) moving to the left
- M leaves a blank (or erases) moving downward
- K leaves a blank (or erases) moving to the right
- P erases the entire screen and is ready to start again

PROGRAM STRUCTURE

- 50-80 initialize coordinates
- 100-520 check for which key was pressed
- 560-690 check to keep writing on screen

RECOMMENDATIONS FOR ALTERATIONS

Have fun! Your imagination is the limit. Fill the screen with your creativity.



```
10 REM U TURN
20 CALL SCREEN(14)
30 CALL CLEAR
40 CALL CHAR(40,"FFFFFFFFFFFFFFFF")
50 XOLD=12
60 YOLD=2
70 X=XOLD
80 Y=YOLD
90 CALL HCHAR(XOLD,YOLD,40)
100 CALL KEY(O,A,STATUS)
110 IF STATUS=0 THEN 100
120 IF A<>ASC("E") THEN 170
130 X=XOLD-1
140 GOSUB 560
150 CALL HCHAR(X,Y,40)
160 GOTO 530
170 IF A<>ASC("X") THEN 220
180 X=XOLD+1
190 GOSUB 560
200 CALL HCHAR(X,Y,40)
210 GOTO 530
220 IF A<>ASC("S") THEN 270
230 Y=YOLD-1
240 GOSUB 560
250 CALL HCHAR(X,Y,40)
260 GOTO 530
270 IF A<>ASC("D") THEN 320
280 Y=YOLD+1
290 GOSUB 560
```

```

300 CALL HCHAR(X,Y,40)
310 GOTO 530
320 IF A<>ASC("I") THEN 370
330 X=XOLD-1
340 GOSUB 560
350 CALL HCHAR(X,Y,32)
360 GOTO 530
370 IF A<>ASC("M") THEN 420
380 X=XOLD+1
390 GOSUB 560
400 CALL HCHAR(X,Y,32)
410 GOTO 530
420 IF A<>ASC("J") THEN 470
430 Y=YOLD-1
440 GOSUB 560
450 CALL HCHAR(X,Y,32)
460 GOTO 530
470 IF A<>ASC("K") THEN 520
480 Y=YOLD+1
490 GOSUB 560
500 CALL HCHAR(X,Y,32)
510 GOTO 530
520 IF A=ASC("P") THEN 30 ELSE 530
530 XOLD=X
540 YOLD=Y
550 GOTO 100
560 IF X<1 THEN 670
570 IF X>24 THEN 650
580 IF Y<1 THEN 630
590 IF Y>32 THEN 610
600 GOTO 690
610 Y=32
620 GOTO 690
630 Y=1
640 GOTO 690
650 X=24
660 GOTO 690
670 X=1
680 GOTO 690
690 RETURN
700 END

```

HANG 'EM HIGH

This program uses more different types of commands than any other program in this book. It is the classic hangman game. You have only ten wrong guesses before the man is completely hanged.

PROGRAM STRUCTURE

120-180	set colors
190-380	define characters
450-470	load an array with ":" to be replaced later with letters guessed
590-620	load an array with words from DATA statements
630-660	randomly choose a word
670	determines the length of the word chosen
680-730	initialize variables used in program
740-830	place gallows on screen
840-1010	place prompts on screen
1020	starts count for ten guesses
1030-1090	accept letter guessed and check to see if it was already guessed
1100	puts letter guessed into an array
1110-1180	check for position of letter in the word and place it on the screen
1190-1230	check to see if letter is in the word
1240	checks to see if word was guessed correctly
1250-1300	control sound if word was guessed
1350-1360	check if there are two or more identical letters in the word
1380-1700	place man on screen
1710-1760	control action if you lose
1770-1780	control counter for wrong guesses
1850	keeps program running

RECOMMENDATIONS FOR ALTERATIONS

You may put your own words into this game by changing the words in the DATA statements. If the total number of words in the DATA statements change, you must change the number "75" in lines 420 and 440 to the new total number of words.



```
100 REM  HANG 'EM HIGH
110 CALL CLEAR
120 CALL SCREEN(14)
130 CALL COLOR(10,7,1)
140 CALL COLOR(11,5,1)
150 CALL COLOR(12,9,1)
160 CALL COLOR(13,15,1)
170 CALL COLOR(14,1,2)
180 CALL COLOR(15,7,2)
190 CALL CHAR(104,"FFFFFFFFFFFFFFFF")
200 CALL CHAR(144,"F7F7F7F7F7F7F7F7")
210 CALL CHAR(136,"F7F7F7F7F7F3F3F3")
220 CALL CHAR(137,"F3EDEDEDEEDF3FF")
230 CALL CHAR(128,"070F1F3F3F3FFFF")
240 CALL CHAR(129,"FEFFFF7C3B170F07")
250 CALL CHAR(130,"E0F0F8FCCECF3FFF")
260 CALL CHAR(131,"7FFFFFF2EDCE8F0E0")
270 CALL CHAR(132,"0303030303030303")
280 CALL CHAR(133,"C0C0C0C0C0C0C0C0C0")
```

```

290 CALL CHAR(112,"FFFFFFFFFFFFFFFF")
300 CALL CHAR(113,"FEFEFEFEFEFEFEFE")
310 CALL CHAR(114,"7F7F7F7F7F7F7F7F")
320 CALL CHAR(110,"000103071F7FFFFFF")
330 CALL CHAR(111,"0080C0E0F8FEFFFF")
340 CALL CHAR(120,"FFFFFFFFFFFFFFFF")
350 CALL CHAR(121,"0103070F1F3F7FFF")
360 CALL CHAR(122,"80C0E0F0F8FCFEFF")
370 CALL CHAR(134,"7FFFFFFEAAAA2828")
380 CALL CHAR(97,"00000000000007E7E")
390 CALL CLEAR
400 REM   LOAD ARRAY
410 OPTION BASE 1
420 DIM WORD$(75)
430 DIM GUES$(25)
440 Z=75
450 FOR W=1 TO 25
460 GUES$(W)=": "
470 NEXT W
480 DATA GYPSIES,PARTIES,BERRIES,HOBBIES,DADDY,STARES
490 DATA BARE,HERE,LOVES,JEWELRY,DESIGNS,ADOBE,LUNCH,HIS
500 DATA BREAKFAST,SUM,FINGERPRINTS,INVESTIGATE,SONS
510 DATA GOLDFISH,FACTORIES,CRIED,TURQUOISE,TUSK
520 DATA FLEW,EAGLE,MACHINE,JUSTICE,AND,BIRD,PULLEY,BITE
530 DATA FORGET,MISTY,CREATURE,MYSTERY,THERMOMETER,TIRED
540 DATA SQUEEZING,ADVENTURE,PLAN,TODAY,MUCH,DEGREE,SKY
550 DATA WORK,SLIPPED,DIRT,TWELVE,LAUGH,ZIPPER,PITCHER
560 DATA DISCOVER,WORD,BOX,DAY,BUSH,SONG,METAL,MAGNET
570 DATA FRIEND,INCH,BUS,DOG,CAT,TOMATO,WEEK,WOOD,BACK
580 DATA BEAR,MUSIC,POP,SEE,MOON,JUNE
590 FOR I=1 TO Z
600 READ A$
610 WORD$(I)=A$
620 NEXT I
630 REM   CHOOSE WORD
640 RANDOMIZE
650 I=INT((Z-1)*RND)+1
660 B$=WORD$(I)
670 A=LEN(B$)
680 X=21
690 Y=4
700 U=0
710 V=0
720 O=25
730 W=1
740 CALL HCHAR(21,12,97,A)
750 CALL VCHAR(2,7,104,15)
760 CALL HCHAR(2,8,104,7)
770 CALL HCHAR(3,11,104)

```

```

780 CALL HCHAR(4,10,104)
790 CALL HCHAR(5,9,104)
800 CALL HCHAR(6,8,104)
810 CALL HCHAR(3,13,136)
820 CALL HCHAR(4,13,137)
830 CALL HCHAR(2,13,144)
840 REM PUT PROMPT ON SCREEN
850 DATA 69,78,84,69,82,32,65,32,76,69,84,84,69,82
860 FOR D=3 TO 16
870 READ N
880 CALL HCHAR(23,D,N)
890 NEXT D
900 CALL HCHAR(2,22,76)
910 CALL HCHAR(2,23,69)
920 CALL HCHAR(2,24,84,2)
930 CALL HCHAR(2,26,69)
940 CALL HCHAR(2,27,82)
950 CALL HCHAR(2,28,83)
960 CALL HCHAR(3,22,71)
970 CALL HCHAR(3,23,85)
980 CALL HCHAR(3,24,69)
990 CALL HCHAR(3,25,83,2)
1000 CALL HCHAR(3,27,69)
1010 CALL HCHAR(3,28,68)
1020 FOR D=1 TO 10
1030 CALL KEY(3,L,ST)
1040 IF ST=0 THEN 1030
1050 L$=CHR$(L)
1060 FOR W=1 TO 25
1070 IF L$=GUES$(W) THEN 1030
1080 NEXT W
1090 U=U+1
1100 GUES$(U)=L$
1110 P=POS(B$,L$,1)
1120 J=0
1130 C=ASC(L$)
1140 IF Y<=24 THEN 1170 ELSE 1150
1150 Y=4
1160 O=27
1170 CALL HCHAR(Y,O,C)
1180 Y=Y+1
1190 IF P=0 THEN 1370
1200 J=1
1210 S=P+11
1220 CALL HCHAR(X,S,C)
1230 V=V+1
1240 IF V=A THEN 1260
1250 GOTO 1350
1260 FOR N=500 TO 1100 STEP 200

```



```

1270 CALL SOUND(50,N,2)
1280 NEXT N
1290 CALL SOUND(70,400,2)
1300 CALL SOUND(300,200,2)
1310 D=20
1320 FOR DELAY=1 TO 1000
1330 NEXT DELAY
1340 GOTO 1780
1350 P=POS(B$,L$,P+1)
1360 GOTO 1190
1370 IF J>0 THEN 1770
1380 ON D GOTO 1390,1440,1470,1500,1540,1580,1600,1620,1660,1690
1390 CALL HCHAR(4,13,128)
1400 CALL HCHAR(5,13,129)
1410 CALL HCHAR(4,14,130)
1420 CALL HCHAR(5,14,131)
1430 GOTO 1780
1440 CALL HCHAR(6,13,132)
1450 CALL HCHAR(6,14,133)
1460 GOTO 1780
1470 CALL VCHAR(7,13,120,3)
1480 CALL VCHAR(7,14,120,3)
1490 GOTO 1780
1500 CALL HCHAR(7,15,120)
1510 CALL HCHAR(7,16,122)
1520 CALL HCHAR(8,16,120)
1530 GOTO 1780
1540 CALL HCHAR(7,12,120)
1550 CALL HCHAR(7,11,121)
1560 CALL HCHAR(8,11,120)
1570 GOTO 1780
1580 CALL HCHAR(9,11,134)
1590 GOTO 1780
1600 CALL HCHAR(9,16,134)
1610 GOTO 1780
1620 CALL HCHAR(10,13,112,2)
1630 CALL VCHAR(11,13,113,2)
1640 CALL VCHAR(11,14,114,2)
1650 GOTO 1780
1660 CALL HCHAR(13,12,110)
1670 CALL HCHAR(13,13,104)
1680 GOTO 1780
1690 CALL HCHAR(13,14,104)
1700 CALL HCHAR(13,15,111)
1710 CALL SOUND(600,440,5)
1720 CALL SOUND(100,310,5)
1730 CALL SOUND(800,110,2)
1740 FOR DELAY=1 TO 1000
1750 NEXT DELAY

```

```
1760 GOTO 1780
1770 D=D-1
1780 NEXT D
1790 IF V=A THEN 1840
1800 CALL CLEAR
1810 PRINT "    YOU LOSE! THE WORD WAS:
           ";WORD$(I)
1820 FOR DELAY=1 TO 1400
1830 NEXT DELAY
1840 RESTORE 480
1850 GOTO 390
1860 END
```


Appendix A

COMMAND GLOSSARY

- ASC** – gives the ASCII character code for the first character of the specified string
- CALL CHAR** – permits you to define your own special graphic characters
- CALL CLEAR** – clears the entire screen
- CALL COLOR** – permits a defined character to be one of sixteen different colors
- CALL HCHAR** – permits a character to be placed on the screen at specified coordinates; if more than one of the same characters is called for, they will be placed next to each other horizontally
- CALL KEY** – permits you to transfer one character from the keyboard directly to your program
- CALL SCREEN** – permits the screen to be one of sixteen colors
- CALL SOUND** – permits the production of different tones and noises
- CALL VCHAR** – permits a character to be placed on the screen at specified coordinates; if more than one of the same characters is called for, they will be placed on the screen next to each other vertically
- CHAR\$** – gives the character for the ASCII character specified
- DATA** – permits storing of information inside your program
- DIM** – reserves spaces for an array
- END** – signifies the completion of your program
- FOR...TO...STEP...NEXT** – permits easy repetition of commands; everything between the FOR and the NEXT is repeated a specified number of times
- GOSUB...RETURN** – permits the completion of lines in another section of the program until the RETURN line is read, then control goes back to the next line immediately after the GOSUB line
- GOTO** – permits you to transfer control of the program to a specified line
- IF...THEN...ELSE** – permits a change in the normal sequence of the program by use of a conditional branch

INPUT - causes the program to pause until data is entered from the keyboard

INT - gives the largest integer of a specified argument

LEN - gives the number of characters in a specified string

NEXT - (see FOR . . . TO . . . STEP . . . NEXT)

ON GOTO - permits the jump to one of several lines depending on a given value

OPTION BASE - allows you to set the lower limit of an array subscript

POS - determines the first occurrence of a specified string within another specified string

PRINT - causes whatever is between the following quotes to be displayed on the screen

RANDOMIZE - when used with RND, it generates a random sequence of numbers each time the program is run

READ - permits access to information stored in DATA statements

REM - permits remarks to be placed into program without affecting the execution of program

RESTORE - permits reuse of DATA statements

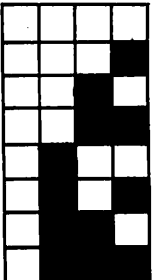
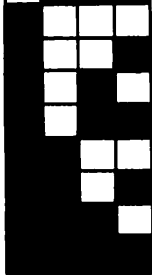


RETURN - (see GOSUB . . . RETURN)

RND - gives the next random number in the current sequence of numbers

Appendix B

HEXADECIMAL CODE

Characters are defined by turning some dots on and other dots off in the grid that makes up each character. This grid consists of 64 dots (8 rows with 8 dots in each row). A binary code is used by the computer to determine which of the dots are on and which are off. We can use a shorter method called the hexadecimal code. All possible combinations of on and off dots, along with their binary and hexadecimal codes, are given below. Note that each row of 8 dots is divided into two sets of 4 dots each when using the codes.

<i>Blocks</i>	<i>Dot Code</i> (0 = off; 1 = on)	<i>Shorthand</i> <i>Code</i>
	0000	0
	0001	1
	0010	2
	0011	3
	0100	4
	0101	5
	0110	6
	0111	7
	1000	8
	1001	9
	1010	A
	1011	B
	1100	C
	1101	D
	1110	E
	1111	F

Appendix C

ASCII CHARACTER CODE

Below is a listing of standard ASCII character codes and character sets.

Code	CHARACTER	Code	CHARACTER
1 {	32 (space)	4 {	56 8
	33 ! (exclamation point)		57 9
	34 " (quote)		58 : (colon)
	35 # (number or pound sign)		59 ; (semicolon)
	36 \$ (dollar)		60 < (less than)
	37 % (percent)		61 = (equals)
	38 & (ampersand)		62 > (greater than)
2 {	39 ' (apostrophe)		63 ? (question mark)
	40 ((open parenthesis)	5 {	64 @ (at sign)
	41) (close parenthesis)		65 A
	42 * (asterisk)		66 B
	43 + (plus)		67 C
	44 , (comma)		68 D
	45 - (minus)		69 E
3 {	46 . (period)		70 F
	47 / (slant)		71 G
	48 0	6 {	72 H
	49 1		73 I
	50 2		74 J
	51 3		75 K
	52 4		76 L
	53 5		77 M
	54 6		78 N
	55 7		79 O

	Code	CHARACTER
7	80	P
	81	Q
	82	R
	83	S
	84	T
	85	U
	86	V
8	87	W
	88	X
	89	Y
	90	Z
	91	[(open bracket)
	92	\ (reverse slant)
	93] (close bracket)
9	94	^ (exponentiation)
	95	— (line)
	96	` (grave)
	97	A
	98	B
	99	C
	100	D
	101	E
	102	F
	103	G

	Code	CHARACTER
10	104	H
	105	I
	106	J
	107	K
	108	L
	109	M
	110	N
11	111	O
	112	P
	113	Q
	114	R
	115	S
	116	T
	117	U
12	118	V
	119	W
	120	X
	121	Y
	122	Z
	123	{ (left brace)
	124	
	125	} (right brace)
	126	~ (tilde)
	127	DEL (appears on screen as a blank)

Appendix D

COLOR CODES

There are 16 colors available on the TI-99/4A. They are used by the CALL COLOR command. An example of this is:

CALL COLOR(4,2,16)

The "4" represents the fourth set of characters (ASCII codes 56-63). The "2" represents the foreground of the characters, which will be black. The "16" represents the background of the characters, which will be white. Below is a listing of the color codes.

COLOR	CODE #
Transparent	1
Black	2
Medium Green	3
Light Green	4
Dark Blue	5
Light Blue	6
Dark Red	7
Cyan	8
Medium Red	9
Light Red	10
Dark Yellow	11
Light Yellow	12
Dark Green	13
Magenta	14
Gray	15
White	16

ART & GRAPHICS WITH YOUR TI-99/4A™

THOMAS A. THOMPSON, JR.

Explore the artistic potential of your computer using 25 ready-to-run programs. All programs are well documented and clearly written in TI BASIC. Introduces a wide range of colorful displays from simple pictures to real animation. Designed for beginners of all ages. You can build programming skills as you create scenic backgrounds and figures. Concludes with illustrations of interactive graphics.

Programs include "Tunnel Vision," "Danger in the Deep," "Fly Trap," "Out of This World," "Old Glory," "Spy in the Sky," "Snake Eyes," "Face to Face," "Play Ball," plus more. All programs can be easily modified and expanded.

Also for TI Home Computers...

FUN AND GAMES WITH YOUR TI-99/4A™

Stephen M. Muncy

Are you ready for the challenge of "Artillery Battle," or "The Last Robot," or "Frog Race"? Arcade-style games test your skills and shape your game-playing abilities in this exciting program collection.

Learn to create your own fun-filled game programs. A step-by-step tutorial with an easy-to-understand review of TI BASIC graphic commands leads you on your way to designing entertaining games. Attractive "mini games" illustrate programming techniques and show you how to get the most from your computer.

Includes "Chopper Rescue," "Wizard Burdoff," "Raceway," and other ready-to-run programs for game players of all ages.



HAYDEN BOOK COMPANY

a division of Hayden Publishing Company, Inc.
Hasbrouck Heights, New Jersey

ISBN 0-8104-6407-1