TI 99/4 A

TIME LOST

A COMPUTER ADVENTURE

que™

Timelost

(Texas Instruments 99/4A Version)

Joseph C. Giarratano
Kris Austen Andrews
Arlan Keith Andrews, Sr.

Series concept: Joseph C. Giarratano and
  Arlan Keith Andrews, Sr.

Story and Artwork: Kris Austen Andrews

Programs: Joseph C. Giarratano

Story consultant: Arlan Keith Andrews, Sr.

Lettering: Andy Andrews

About the Authors

Arlan Keith Andrews, Sr., is a mechanical engineer at American Bell in Indianapolis. He received his Doctor of Science in Engineering degree from New Mexico State University and is a member of Science Fiction Writers of America. Kris Andrews, born in 1967 at Las Cruces, New Mexico, is currently a junior at Carmel High School, Carmel, Indiana. He is an honor student who desires to be a theoretical physicist. He is also a science fiction fan and likes role-playing games and computers. Joseph C. Giarratano received his B.S. and M.S. degrees in physics from California State University at Los Angeles and his Ph.D. degree in physics from the University of Texas at Austin. He is the author of: *Foundations of Computer Technology; Modern Computer Concepts; BASIC: Fundamental Concepts; BASIC: Advanced Concepts;* the *Timex/Sinclair 1000 User's Guide,* Volumes 1 and 2; and the *Timex/Sinclair 1000 Dictionary and Reference Guide.*

Acknowledgment

Dedications

To my family:
Jane, Jenna, Melissa, and Anthony Giarratano
—*Joseph Giarratano*

To the family:
Joyce, Patty, Andy, Mandy, Sean,
Vivian Andrews Weed, Arlie Andrews,
and Mrs. Lizzie Van Ness
—*Kris and Arlan Andrews*

...and my Mom
—*Kris Andrews*

Other *Timelost* Books by Que

Book:	ISBN No.	Date Available
Timelost (Atari Version)	0-88022-056-2	July, '83
Timelost (Timex/Sinclair 1000 Version)	0-88022-030-9	June, '83
Timelost (VIC 20 Version)	0-88022-054-6	July, '83

Introduction

Welcome to the Universe of THE TIMELOST! In this new and exciting series, you'll follow the adventures of John Moore, a teenage computer whiz; his little sister, Erin; and their mysterious friend from the future. Join them as they battle against the evil Wizard, a Warlord of the Universe.

These books combine a story with programs that you can enter and run on a standard Texas Instruments 99/4A computer with 16K bytes of memory. You follow the story and then get into the action with the game programs. All of the games were written especially to follow the adventures in the book.

These game programs are designed to be fun and educational. If you don't know how to program, you'll enjoy playing the games and may wish to learn about programming. If you do know how to program, or are learning how, the detailed explanations will show you how the games work and how you can change them and improve them.

The games in this book are designed to illustrate action and cover many different

sound, color, and animation effects you can achieve with your computer. We know they'll be fun!

So, enter the fun world of computer games and the mysterious worlds of THE TIMELOST....

NOTE: In the game programs, a triangle means a space. Also the numbers in parentheses in the program explanations represent the line numbers on which the related program segment is entered.

Note to the Reader

   Timelost is designed so that it can be used in different ways. The cartoons can be read straight through as an episodic adventure story. Or, if you wish, you can choose the games that appeal to you the most, key the programs into your computer, and just play those games. Each game is self-contained. You don't have to read the cartoon section to understand how to play the game.

   You may get the most enjoyment, however, by reading one episode and then playing the corresponding game at the back of the book. If you want to use the book this way, look for the "Play the game here" notation in the bottom right-hand corner of certain pages. At that point, simply refer to the Contents at the beginning of the book to find the page on which the appropriate game begins.

   All game program listings have been thoroughly tested by the programmer(s) and work properly if typed correctly. If a problem occurs in playing any of the games, check carefully your entry of the listing.

TIMELOST GAME CASSETTES!

For instant enjoyment of the Timelost game programs, order the Timelost cassette tapes for your computer. Each tape contains the programs exactly as they are found in the book. Play the computer games with all your favorite Timelost characters—without having to enter the programs yourself!

Please send me the following Timelost cassette(s):

Description	How Many?
Timelost cassette, VIC-20 format	_____
Timelost cassette, TI-99/4A format	_____
Timelost cassette, Atari format	_____

_____ cassettes x $19.95 each = $_____

Shipping & Handling ($2.00 per item) = $_____

TOTAL = $_____

Method of Payment:

Check_____ MasterCard_____ VISA_____ Am. Exp._____

Credit Card Number: _____

Expiration Date: _____

Cardholder Name: _____

Ship To:_____

Address: _____

City: _____ State:_____ Zip:_____

Tear out and mail to:
     Timelost Game Cassette
     Que Corporation
     7960 Castleway Drive
     Indianapolis, IN 46250

Contents

Games *(cont'd)*

WE ARE STANDING AT THE EDGE OF THE LARGEST OF SEVERAL CIRCLES OF STONE... "MEGALITHS", WE CALL THEM. THE TALLEST STONES ARE TWENTY FEET HIGH, IN A CIRCLE 108 FEET IN DIAMETER. FURTHER OUT, SMALLER STONES LINE UP WITH THESE CIRCLES. SOME SCIENTISTS SAY THAT THE ALIGNMENTS POINT OUT THE BEGINNINGS OF THE SEASONS, AND THE RISING OF THE STARS.

WHATEVER THEY WERE USED FOR, THEY ARE FAR OLDER THAN THE PYRAMIDS OF EGYPT.

THE LATEST **MODERN** EXPLANATION, TAKING INTO ACCOUNT ALL OF THE CIRCLE ALIGNMENTS, IS THAT STONEHENGE IS ACTUALLY AN ASTRONOMICAL **COMPUTER**-- A COMPUTER THAT WAS USED TO PREDICT ECLIPSES AND PHASES OF THE MOON WITH ALMOST THE SAME PRECISION AS THE LATEST COMPUTERS OF OUR TIME.

IN THE TOUR GROUP, DRS. MARY AND PETER MOORE, COMPUTER SCIENTISTS, LISTEN INTENTLY, UNAWARE OF THE IMPENDING FATE OF THEIR CHILDREN...

THERE IS A LESS PLAUSIBLE, BUT MUCH MORE ROMANTIC, LEGEND WHICH CLAIMS THAT IN KING ARTHUR'S TIME, THE STONES WERE PUT IN PLACE BY A MAGICIAN...
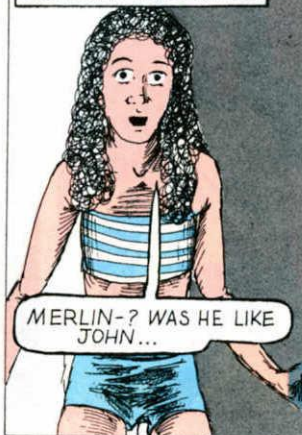
... TEENAGER JOHN MOORE, HIGH SCHOOL COMPUTER WHIZ...

WOW!

... A SORCERER WE KNOW AS MERLIN.

...AND SEVEN-YEAR-OLD ERIN MOORE, DEVOTEE OF COMPUTER VIDEO GAMES.

MERLIN-? WAS HE LIKE JOHN...

A NIGHTMARE ATTACKER...

A MYSTERIOUS STRANGER...

CONFUSION RUNS RAMPANT, BUT ONE THING IS CLEAR---
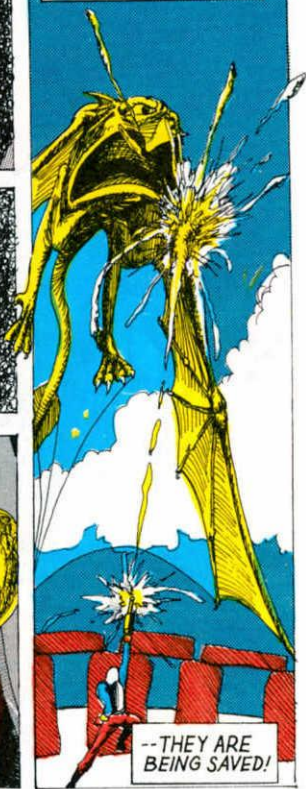
--THEY ARE BEING SAVED!
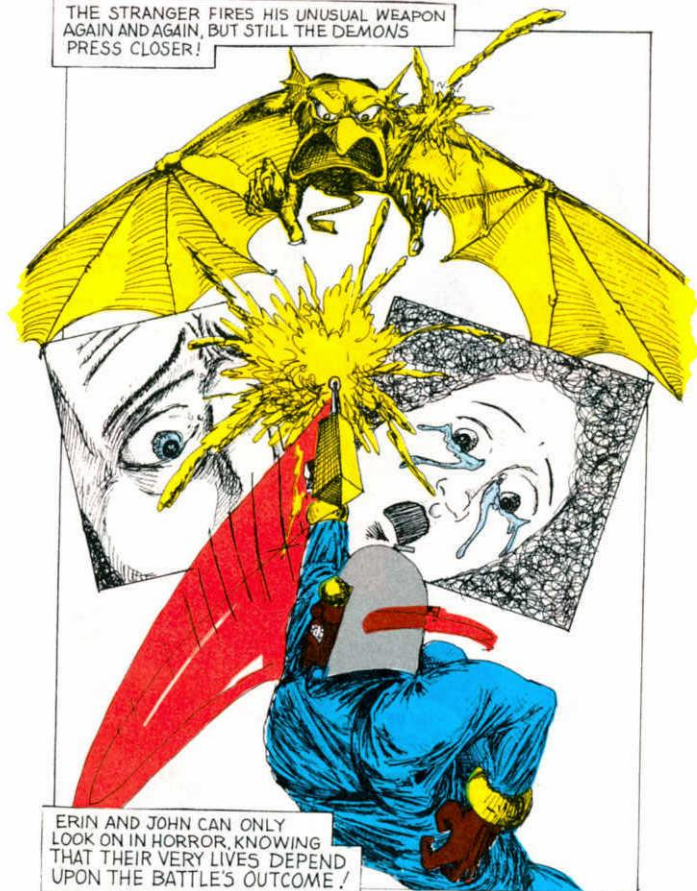
Play the game here.

IT IS ANOTHER PLACE, ANOTHER TIME. THE DEMON IS GONE; THE MYSTERY OF STONEHENGE SEEMINGLY FAR BEHIND, REPLACED BY YET A DEEPER ENIGMA:

HOW, AND MORE IMPORTANTLY **WHY**, WERE THEY BROUGHT TO THIS RANK AND DISMAL PLACE IN A FORGOTTEN CORNER OF TIME?

J-JOHN--? WH-WHERE ARE WE--?

THE SEETHING SWAMP SEEMS TO STRETCH FARTHER THAN THE YOUTHS' STUNNED MINDS CAN ACCEPT. AS FOR THE STRANGER, HIS THOUGHTS ARE HIS OWN.

Play the game here.

WITHOUT WARNING THE VERY GROUND CAVES IN ON ITSELF, OPENING A PIT WITHOUT DEPTH...

...UNLEASHING A SAVAGE SWARM OF TUNNELLING TERRORS BENT ON ONE PUROSE:

GET BACK!

...THE ABDUCTION OF THE UNLIKELY QUARTET!
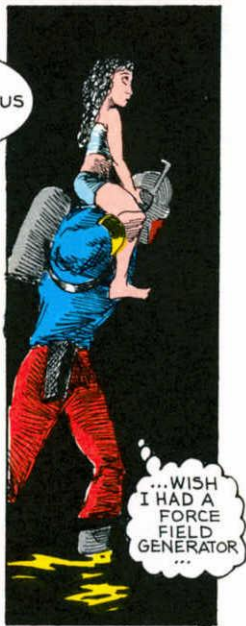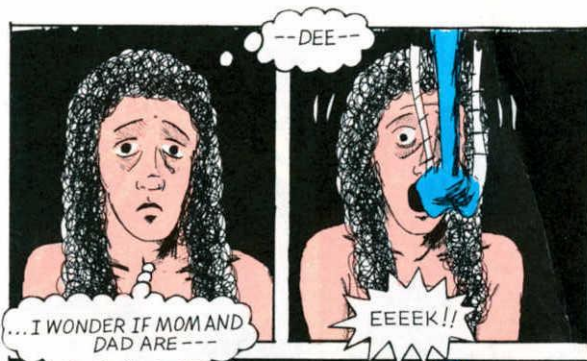
WELL, THAT'S WHERE THE SIGNAL STOPPED, SO THAT'S WHERE I'M GOING!

ERIN, YOU STAY HERE WITH D'REEN...

ER- I CAN'T WATCH HER! I CAN BARELY LOOK AFTER MYSELF IN THIS PLACE!

BESIDES, I WANNA STAY WITH YOU, JACQUE! I WANNA HELP FIND JOHN!

IT'LL BE DANGEROUS--- BUT OKAY. BUT YOU'LL HAFTA KEEP QUIET, AND YOU'LL HAFTA DO WHAT I SAY, OR ELSE I'LL LEAVE YA THERE TA DIE AND FIND MY OWN WAY HOME! GOT THAT?

Y-YESSIR...
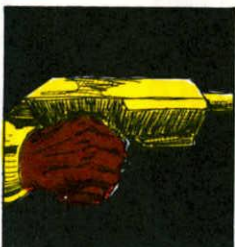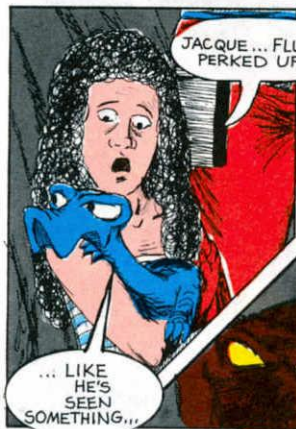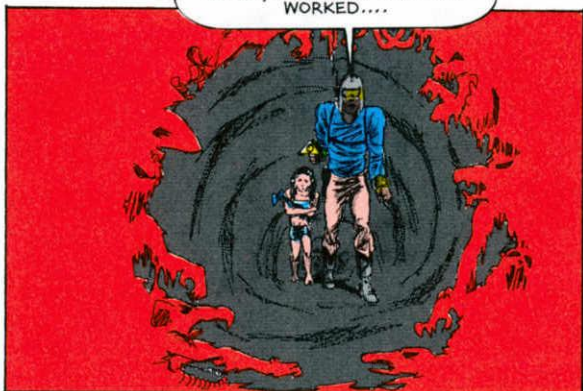
...AND THE FRAY IS ENTERED! THE CAVERNOUS HALL IS FILLED WITH THE INSANE HOWLS OF THE ONCOMING GHOULS, THE THUNDEROUS BOOMS OF THE POSITRON BLASTER, AND THE ECHOING SCREAMS OF A HORRIFIED CHILD!!

Play the game here.

THOUGH WE SHALL SOON FIND OUT...

WE OF THE VILLAGE MARSHSIDE, THEREFORE, HAVE TAKEN UPON OURSELVES TO SHELTER TRAVELLERS FROM THE WIZARD'S CLUTCHES, AND AID THEM IN THEIR RETURNS HOME. BUT ALAS...

---ALAS, THE LAST FOOL WE GAVE OUR AID TO BETRAYED US AND TURNED HIMSELF INTO THE LORD-WIZARD, DESPITE OUR WARNINGS. SOON AFTERWARD WE FOUND OURSELVES ATTACKED BY SNYARLIMHOTHEP'S UNEARTHLY MINIONS AND DRUG HERE TO GODS KNOW WHAT FATE!

Play the game here.

NOW, ERIN, **NOW**!!

THE CAVE WALLS BUCKLE UNDER THE FORCE OF THE BLOW---
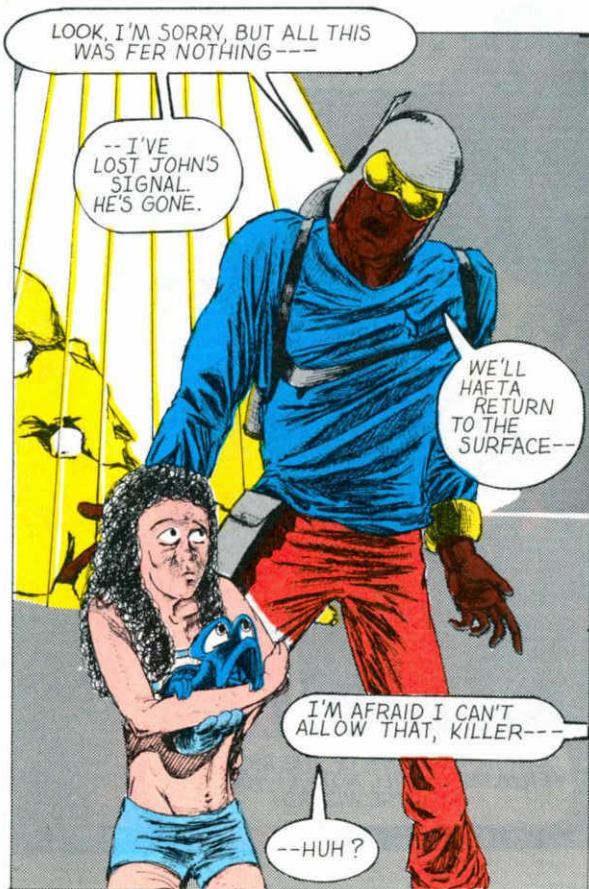
---AND ONLY SECONDS WILL TELL IF ERIN CAN PERFORM *HER* TASK RIGHT!

Battle at Stonehenge

How long can you defy the terrible Warbirds of the Wizard? Test your skill with a positron beam as you fight off the dreaded Warbirds in the Battle of Stonehenge. But beware: if a Warbird reaches the ground, the game is over.

Enter the following program and run it. You control the mystery man at the bottom of the screen. Press the "S" key to move left and the "D" key to move right. Use the space bar to fire his Positron Blaster when he is directly under the Warbird! To get more points, wait until the Warbird is as close as possible to the Mystery Man.

GAME PROGRAM LISTING

```
10 CALL CLEAR
20 CALL SCREEN(11)
30 PRINT "BATTLE AT STONEHENGE"
40 PRINT :::"PRESS S (LEFT) OR D (RIGHT)
   TO MOVE."
50 PRINT :::"PRESS THE SPACEBAR TO FIRE."
60 PRINT :::"PRESS ANY KEY TO BEGIN."
70 CALL KEY(0,K,S)
80 IF K<0 THEN 70
```

```
 90 REM **BATTLE AT STONEHENGE
100 CALL SCREEN(2)
110 CALL CLEAR
120 RESTORE
130 RANDOMIZE
140 REM **ASSIGN VARIABLES
150 SCORE=0
160 LL=20
170 RR=15
180 IF TA=2 THEN 630
190 REM **CHAR DESIGN
200 CALL CHAR(144,"3C995A3C18183C18")
210 CALL CHAR(40,"0")
220 CALL CHAR(104,"0")
230 CALL CHAR(154,"1898FF3D3C3CE404")
240 CALL CHAR(155,"1819FFBC3C3C2720")
250 CALL CHAR(120,"1818181818181818")
260 CALL CHAR(133,"FFFFFFFFFFFFFFFF")
270 CALL CHAR(134,"00000000FFFFFFFF")
280 CALL CHAR(135,"3C3C3C3C3C3C3C3C")
290 CALL CHAR(145,"3C18183C5A993C18")
300 REM **COLOR
310 CALL COLOR(8,8,8)
320 CALL COLOR(4,2,8)
330 CALL COLOR(15,16,13)
340 CALL COLOR(2,4,13)
350 CALL COLOR(16,2,13)
```

```
360  CALL COLOR(13,2,2)
370  CALL COLOR(12,12,13)
380  REM **PILLARS
390  CALL HCHAR(16,4,133,5)
400  CALL HCHAR(16,25,133,5)
410  READ AA
420  IF AA=0 THEN 470
430  FOR J=17 TO 20
440  CALL HCHAR(J,AA,133,1)
450  NEXT J
460  GOTO 410
470  READ BB
480  CALL HCHAR(6,6,134,5)
490  CALL HCHAR(6,23,134,5)
500  IF BB=0 THEN 550
510  FOR J=7 TO 10
520  CALL HCHAR(J,BB,135,1)
530  NEXT J
540  GOTO 470
550  CALL COLOR(10,4,10)
560  FOR JJ=1 TO 5
570  FOR II=1 TO 5
580  CALL HCHAR(JJ,II,104,28)
590  NEXT II
600  NEXT JJ
610  DATA 5,7,26,28,0
620  DATA 7,9,24,26,0
```

```
630 REM **SCORE BOARD
640 X$="SCOREZ=ZZZZZZ"
650 FOR B=1 TO LEN(X$)
660 CALL HCHAR(3,B+10,ASC(SEG$(X$,B,1)))
670 NEXT B
680 CALL SCREEN(8)
690 CALL COLOR(1,13,13)
700 CALL COLOR(13,2,13)
710 REM **INITIAL POSITIONS
720 R=INT(12*RND)+10
730 SH=INT(4*RND+6)
740 FOR CL=SH TO 20
750 IF CL=20 THEN 1220
760 CALL HCHAR(LL,RR,154)
770 CALL HCHAR(CL,R,144)
780 REM **DECISIONS
790 CALL KEY(0,K,C)
800 IF K=83 THEN 920
810 IF K=68 THEN 860
840 IF K=32 THEN 1040 ELSE 1000
850 REM **MOVEMENTS
860 CALL HCHAR(LL,RR,40)
870 RR=RR+1
880 IF RR>21 THEN 930 ELSE 990
920 CALL HCHAR(LL,RR,40)
930 RR=RR-1
940 IF RR<10 THEN 870 ELSE 990
```

```
 980  GOTO 1000
 990  CALL HCHAR(LL,RR,155)
1000  CALL VCHAR(CL,R,40)
1010  CALL VCHAR(CL+1,R,145)
1020  CALL SOUND(100,30*CL,21-CL)
1030  NEXT CL
1040  REM **POSITRON BLAST
1050  LK=LL-1
1060  FOR ML=LK TO 6 STEP -1
1070  CALL HCHAR(ML,RR,120)
1080  IF ML=CL THEN 1090 ELSE 1100
1090  IF RR=R THEN 1140
1100  CALL HCHAR(ML,RR,40)
1110  IF CL>ML THEN 1000
1120  NEXT ML
1130  GOTO 1000
1140  CALL HCHAR(ML,RR,40)
1150  CALL SOUND(100,-7,0)
1160  CALL SOUND(100,-7,5)
1170  SCORE=SCORE+CL
1180  FOR BB=1 TO LEN(STR$(SCORE))
1190  CALL HCHAR(3,BB+17,ASC(SEG$(STR$
      (SCORE),BB,1)))
1200  NEXT BB
1210  GOTO 720
1220  REM **TRY AGAIN
1230  TA=2
```

```
1240 X$="ANOTHER GAME?"
1250 FOR BB=1 TO LEN(X$)
1260 CALL HCHAR(24,BB+3,ASC(SEG$(X$,BB,1)))
1270 NEXT BB
1280 CALL KEY(0,K,C)
1290 IF K=89 THEN 1300 ELSE 1340
1300 CALL HCHAR(24,4,32,14)
1310 CALL HCHAR(LL,RR,32)
1320 CALL HCHAR(CL,R,32)
1330 GOTO 140
1340 IF K=78 THEN 1350 ELSE 1280
1350 END
```

Variables Used in the Game Include:

K=A key that has been pressed
LL=The starting column of the Mystery Man
RR=The starting row of the Mystery Man
TA=Whether the game has been played
AA=The columns of the pillars
J=The rows of the pillars
BB=The columns of the pillars
JJ=The rows of the sky
II=The columns of the sky
B=The beginning column of a statement
X$=Statements used in the game
R=The column of the Warbird
SH=The beginning row of the Warbird
CL=The changing row of the Warbird
LK=The starting row of the Positron Blast
ML=The changing row of the Positron Blast

BB=The beginning column of a statement

The colors used in this game are as follows: Mystery Man— black; the Warbird—white; and the Positron Blast—black.

The Mystery Man moves by pressing the "S" (left) or "D" (right). The space bar controls the Positron Blast.

Explanation of the Program

The screen clears (10) and changes to a light yellow (20). The name of the game (30) and directions (40-60) are displayed until the player presses any key (70-80).

This section causes the screen to become black (100), clears any characters from the screen (110), allows data statements to be used again (120), and tells the computer to select numbers randomly (130).

The score is set to zero (150), the row number is set to 12 (160), the column number is set to 12 (170), and a check is made to determine if the game has been played (180).

Next are designed the Mystery Man (230, 240); the Warbird (200, 290); the Positron Blast (250); the space that replaces the man, the Warbird, and the blast (210); and the pillars (220, 260, 270). Color is assigned to the various characters: the mystery man (350), the blast (370), the Warbird (330), and the pillars (360).

The tops of the pillars are drawn (390-400), and the "legs" are read from data (410, 610) and

drawn (430-450). When "0" is read (420), the tops of the distant pillars are drawn (390-400), and the "legs" are read (470, 620) and drawn (500-530). When "0" is read (500), the sky is given color (550) and drawn (560-600).

The scoreboard is drawn next. X$ contains 13 characters. This means that LEN(X$) is 13. Therefore, the lines are repeated 13 times.

Now that the pillars have been drawn, the screen (680), spaces (690), and pillars (700) are given new colors.

The Warbird's position is randomly chosen (710-720), and the Warbird (760) and the Mystery Man (770) are drawn.

These lines control movements of both creatures and the Positron Blast. While the Warbird is moving down the screen, the computer waits for the player to do something (790-840). If nothing happens, the computer blanks out the Warbird (1000), draws it one row closer to the Mystery Man (1010), makes a sound that reflects its position on the screen (1020), goes to line 740, and repeats the procedure again.

If the player decides to move (800-810), lines 920-940 are executed to move the Mystery Man left, and lines 860-880 are executed to move the Mystery Man right.

If the player decides to shoot, the blast is drawn one row above the Mystery Man (1000-1020), is checked for a hit (1080-1090), and is

blanked out if it misses the bird (1100-1120). If the blast misses, or if it runs out of energy, the procedure is repeated again (1130).

If the blast hits the bird, the blast and the bird are blanked out (1140); sound effects are played (1150-1160); the score is increased by the row number of the Warbird (1170) and drawn (1180-1200); and a new position is chosen for the bird, and the procedure is repeated (1210).

If the Warbird moves past the Mystery Man, the game is over (750). A variable is reset (1230) to indicate that the game has been played one time. A question is drawn (1240-1270), and the computer waits for the player to answer (1280-1290). If the player wants to play again, the game is restarted (1300-1330). Otherwise, the game ends (1350).

Some Changes You Might Like to Try:

1. Make the Warbird move in a zigzag pattern down the screen, instead of straight down.
2. Penalize the player for missed shots. For example, give 10 points for every Warbird the man hits and take away 5 points for every one missed.
3. Allow multiple Warbirds on the screen, instead of just one at a time.
4. Allow the Warbird to descend faster. Hint: look at line 1010.

Attack of the Slime Creatures

Hold off the Slime Creatures! How long can you avoid the terrors as they come closer and closer?

The Slime Creatures appear and disappear as they swim underwater toward Jacque. Then suddenly they appear for a moment as they surface to attack Jacque.

Shoot at them with Jacque's gun by pressing the space bar, and score more points the closer they are. You'll hear a sound if they're hit. Otherwise, the Positron Blast will have missed because they have dived into the swamp before being hit. But watch out! If you miss, the Slime Creatures may randomly move up to twice as far compared to when you don't fire. Your only clue as to how long they'll appear is a series of beeps before they surface as their air bubbles rise to the surface. The number of beeps is proportional to how long they'll be visible. Also, as they come closer, the pitch and intensity of the sounds increase.

GAME PROGRAM LISTING

```
 10 CALL CLEAR
 20 CALL SCREEN(11)
 30 PRINT "ATTACK OF THE SLIME CREATURES."
 40 PRINT :::"PRESS THE SPACE BAR TO FIRE."
 50 PRINT :::"PRESS ANY KEY TO BEGIN."
 60 CALL KEY(0,K,S)
 70 IF K<0 THEN 60
 80 REM SLIME CREATURES
 90 CALL CLEAR
100 REM **ASSIGN VARIABLES
110 I=5
120 SCORE=0
130 IF TA=2 THEN 370
140 REM **CHAR DESIGN
150 CALL CHAR(154,"1898FF3D3C3CE404")
160 CALL CHAR(145,"111F3766FE9E1F1F")
170 CALL CHAR(144,"3FFEFE7E1E1E1E1E")
180 CALL CHAR(128,"000000FFFF")
190 REM **COLOR
200 CALL COLOR(14,5,5)
210 CALL COLOR(11,5,2)
220 CALL COLOR(16,2,9)
230 CALL COLOR(8,4,4)
240 CALL COLOR(2,7,7)
250 CALL COLOR(13,11,5)
260 CALL COLOR(1,9,9)
```

```
270  CALL COLOR(15,12,5)
280  REM **SWAMP
290  FOR JJ=24 TO 9 STEP -1
300  CALL HCHAR(JJ,I,136,32-I)
310  X=X+1
320  IF X>3 THEN 330 ELSE 350
330  I=I+1
340  X=0
350  NEXT JJ
360  REM **SCOREBOARD
370  X$="SCOREZ=        "
380  FOR B=1 TO LEN(X$)
390  CALL HCHAR(4,B+3,ASC(SEG$(X$,B,1)))
400  NEXT B
410  REM **INITIAL POSITIONS
420  CALL HCHAR(14,6,32)
430  CALL HCHAR(15,6,154)
440  M=INT(18*RND)+15
450  M=M-INT(4*RND)-1
460  FOR JO=1 TO 10*M-70
470  NEXT JO
480  IF M<7 THEN 840
490  TT=RND*M
500  CALL HCHAR(15,M,136)
510  CALL HCHAR(14,M,136)
520  FOR JJ=1 TO TT
530  CALL SOUND(200,2100-60*M,M-7)
```

```
540 NEXT JJ
550 CALL HCHAR(15,M,144)
560 CALL HCHAR(14,M,145)
570 FOR J=1 TO TT
580 CALL KEY(0,K,C)
590 IF K=32 THEN 650
600 NEXT J
610 CALL HCHAR(15,M,136)
620 CALL HCHAR(14,M,136)
630 GOTO 450
640 REM **POSITRON BLAST
650 FOR RR=7 TO M
660 TT=TT-1
670 IF TT<=0 THEN 970
680 CALL HCHAR(15,RR,128)
690 CALL HCHAR(15,RR,136)
700 CALL HCHAR(14,RR,136)
710 IF RR=M THEN 740
720 NEXT RR
730 REM **SCORE
740 CALL HCHAR(15,RR,136)
750 CALL HCHAR(14,RR,136)
760 CALL SOUND(100,-7,0)
770 CALL SOUND(100,-7,0)
780 SCORE=SCORE+(30-M)
790 FOR B=1 TO LEN(STR$(SCORE))
```

```
 800 CALL HCHAR(4,B+11,ASC(SEG$(STR$
     (SCORE),B,1)))
 810 NEXT B
 820 GOTO 440
 830 REM **TRY AGAIN
 840 TA=2
 850 CALL HCHAR(15,6,144)
 860 CALL HCHAR(14,6,145)
 870 X$="ANOTHER GAME?"
 880 FOR B=1 TO LEN(X$)
 890 CALL HCHAR(6,B+6,ASC(SEG$(X$,B,1)))
 900 NEXT B
 910 CALL KEY(0,K,C)
 920 IF K=89 THEN 940
 930 GOTO 960
 940 CALL HCHAR(6,3,32,32)
 950 GOTO 110
 960 IF K=78 THEN 1150 ELSE 910
 970 CALL HCHAR(15,M,136)
 980 CALL HCHAR(14,M,136)
 990 CALL HCHAR(15,RR,136)
1000 CALL HCHAR(14,RR,136)
1010 MOLD=M
1020 M=M-INT(8*RND)-1
1030 IF M<7 THEN 840
1040 FOR J=RR TO MOLD
1050 CALL HCHAR(15,J,128)
```

The colors in this game are as follows: Jacque—black; the Slime Creatures—white; the Blast—light yellow.

The Positron Blaster is controlled by pressing the space bar. You can only shoot after the creature appears.

Explanation of the Program

The screen clears (10) and changes to a light yellow (20). The name of the game (30) and directions (40-50) are displayed until the player presses any key (60-70).

The screen clears (90), and several variables are set: the starting column number of the swamp (110); then the score, which is set to zero (120). A test statement (130) allows the player to by-pass lines 140-360. This speeds up the time between plays.

Next Jacque (150), the Slime Creatures (160-170), and the blast (180) are designed.

Colors are given to Jacque (220), the Slime Creatures (270), the blast (250), a space (200), and letters and numbers (210, 230, 240, 260).

The swamp is drawn (290-320). Each time a row is filled, a counter is increased by 1 (310). When the counter becomes greater than 3 (320), the column number is increased by 1 (330), the first counter is set to zero (340), and the swamp's next row is drawn again. This continues until the loop is completed (280-350).

```
1060  CALL HCHAR(15,J,136)
1070  IF J<>M+1 THEN 1130
1080  CALL HCHAR(15,M,144)
1090  CALL HCHAR(14,M,145)
1100  FOR JJ=1 TO 3
1110  CALL SOUND(200,2100-60*M,M-7)
1120  NEXT JJ
1130  NEXT J
1140  GOTO 490
1150  END
```

Variables Used in this Game Include:

K=A key that has been pressed

I=The starting column of the swamp

TA=Whether the game has been played

JJ=The rows of the swamp

X=The number of times a row has been printed (The column is adjusted every time X=3.)

X$=Various statements used in the game

B=The column where a statement will be drawn

M=The column of a Slime Creature

JO=The time a Slime Creature remains hidden

TT=The amount of time a creature's head is above water

RR=The column of the Positron Blast

MOLD=Old value of the column where the Slime Creature was before it disappeared under the water

The scoreboard is drawn (360-400).

Jacque is drawn (430), and a position is randomly chosen for the creature (440, 450). A delay (460-470) is set to correspond to the creature's position. If the Slime Creature's position is less than 7 (480), the game ends because the hero is in column 6.

A number is chosen (490) that reflects the amount of time the creature's head will appear on the screen (550-560). In lines 520-540, a number of beeps are sounded proportional to TT, the amount of time that the Slime Creature will appear. The pitch and loudness of the sound increase as the creature gets closer to the hero.

While the action is continuing on the screen, the computer waits for the player to shoot (590). If nothing happens, the computer blanks out the Slime Creature (610-620) and chooses a new starting position for the Slime Creature (450).

When the player presses the space bar (590), a blast is drawn one space beyond Jacque (650). The time is begun that shows the head of the creature (660) and checked to see if time has run out (670). The blast is then drawn (680) and blanked out (690). If the blast hits the Slime Creature while its head is up, a hit is scored (710). If not, the blast continues (720).

The blast is blanked out (740) and sounds are made (760-770). The score is increased by the column number of the Slime Creature (780) and drawn on the screen (790-810). A new position is then chosen for the Slime Creature (820).

When the creature reaches Jacque, a counter is set to signify that the game has been played one time (840). A message is drawn (870-900), and the computer waits for the player to decide either to play again (920) or quit (960). If the player decides to play again, the message is blanked out (940) and the counter is reset (950). The test statement is checked (130), and the game begins again. Otherwise, the game ends (1150).

Some Changes You Might Like to Try:
1. Allowing multiple Slime Creatures to appear and disappear. You would allow Jacque to move on the screen to position him under the creature you want to shoot.
2. Limiting the number of Positron Blasts and Slime Creatures per game. For example, allow 20 Slime Creatures and 20 Positron Blasts to see if Jacque can hit them all. You should also display the Positron Blasts remaining.

Peril of
the Pitdemons

Now, you must fight off a fierce attack by underground creatures of the Wizard! In this game, you control Jacque's blaster as he fights against the swarming demons from the dark pit.

You move Jacque, using the "S" control to go left and the "D" control to move right. Use the space bar to fire the Positron Blaster. Your score is shown on the left of the screen. You can play until a Pitdemon reaches the top row, and then the game ends. You get more points the closer you let the Pitdemons approach Jacque.

GAME PROGRAM LISTING

```
10  CALL CLEAR
20  CALL SCREEN(11)
30  PRINT "PERIL OF THE PITDEMONS"
40  PRINT :::"PRESS S (LEFT) OR D (RIGHT)
    TO MOVE."
50  PRINT :::"PRESS THE SPACE BAR TO FIRE."
60  PRINT :::"PRESS ANY KEY TO BEGIN."
```

```
 70  CALL KEY(0,K,S)
 80  IF K<0 THEN 70
 90  REM **PITDEMONS
100  CALL CLEAR
110  RANDOMIZE
120  LL=16
130  IF TA=2 THEN 580
140  CALL SCREEN(2)
150  REM **CHAR DESIGN
160  CALL CHAR(152,"1819FF3D3C3C2770")
170  CALL CHAR(153,"1819FFBC3C3C4280")
180  CALL CHAR(126,"183D7EBC3C3D4280")
190  CALL CHAR(127,"18BC7E3D3CBC4201")
200  CALL CHAR(104,"1818181818181818")
210  CALL CHAR(96,"FFFFFFFFFFFFFFFF")
220  REM **ASSIGN COLOR
230  CALL COLOR(2,2,2)
240  CALL COLOR(5,5,2)
250  CALL COLOR(6,8,2)
260  CALL COLOR(8,9,2)
270  CALL COLOR(3,7,2)
280  CALL COLOR(4,10,2)
290  CALL COLOR(10,16,2)
300  CALL COLOR(9,12,12)
310  CALL COLOR(11,12,12)
320  CALL COLOR(12,7,2)
330  CALL COLOR(13,11,4)
```

```
340  CALL COLOR(14,16,4)
350  CALL COLOR(15,5,4)
360  CALL COLOR(16,6,12)
370  CALL COLOR(7,4,2)
380  REM **ROCKS
390  FOR J=1 TO 4
400  CALL HCHAR(J,1,112,32)
410  NEXT J
420  REM **PIT
430  READ P
440  IF P=0 THEN 530
450  READ U,C1
460  CALL HCHAR(P,U,112,C1)
470  GOTO 430
480  DATA 5,1,10,5,22,10,6,1,10,6,22,10,7,1,9,7,24,8,
     8,1,9,8,24,8
490  DATA 9,1,7,9,26,6,10,1,7,10,26,6
500  DATA 11,1,5,11,27,5,12,1,5,12,28,4,13,1,3,13,
     30,2,14,1,3,14,30,2
510  DATA 15,1,2,15,30,2,16,1,2,16,30,2,17,1,4,17,
     30,2,18,1,4,18,30,2,19,1,6,19,28,4,20,1,6
520  DATA 20,28,4,21,1,8,21,25,7,22,1,9,22,25,7,23,
     1,10,23,24,8,24,1,32,0
530  REM **SCOREBOARD
540  X$="SCORE = "
550  FOR BB=1 TO LEN(X$)
560  CALL HCHAR(1,BB+3,ASC(SEG$(X$,BB,1)))
```

```
570 NEXT BB
580 CALL HCHAR(4,LL,152)
590 REM **SET DIM
600 DIM AA(7,2)
610 FOR J=1 TO 7
620 FOR I=1 TO 2
630 READ AA(J,I)
640 NEXT I
650 NEXT J
660 DATA 22,19,22,16,22,15,22,18,22,14,22,13,22,17
670 IF XX=0 THEN 680 ELSE 690
680 GOSUB 890
690 REM **DECISIONS
700 CALL KEY (0,K,S)
710 IF K=32 THEN 1120
720 IF K=83 THEN 770
730 IF K=68 THEN 820
740 GOSUB 890
750 GOTO 690
760 REM **HERO MOVE
770 CALL HCHAR(4,LL,96)
780 LL=LL-1
790 IF LL<11 THEN 830
800 CALL HCHAR(4,LL,153)
810 GOTO 860
820 CALL HCHAR(4,LL,96)
830 LL=LL+1
```

```
 840 IF LL>21 THEN 780
 850 CALL HCHAR(4,LL,153)
 860 GOSUB 890
 870 GOTO 690
 880 REM **PITDEMON MOVE
 890 IF XX=0 THEN 950 ELSE 900
 900 M=INT(RND*7)+1
 910 A=AA(M,1)
 920 B=AA(M,2)
 930 CALL HCHAR(A,B,32)
 940 IF XX=1 THEN 960
 950 FOR M=1 TO 7
 960 AC=AA(M,1)
 970 AB=INT(RND*3)+1
 980 AC=AC-AB
 990 B=AA(M,2)
1000 IF AC<7 THEN 1010 ELSE 1040
1010 CALL HCHAR(4,B,126)
1020 AA(M,1)=5
1030 GOTO 1370
1040 CALL HCHAR(AC,B,126)
1050 CALL SOUND(100,600-20*AC,AC)
1060 AA(M,1)=AC
1070 IF XX=1 THEN 1100
1080 NEXT M
1090 XX=1
1100 RETURN
```

```
1110 REM **POSITRON BLAST
1120 FOR J=1 TO 7
1130 IF LL=AA(J,2)THEN 1160
1140 NEXT J
1150 J=0
1160 FOR LK=5 TO 21
1170 CALL HCHAR(LK,LL,104)
1180 CALL HCHAR(LK,LL,32)
1190 IF J=0 THEN 1210
1200 IF LK=AA(J,1)THEN 1240
1210 NEXT LK
1220 GOSUB 890
1230 GOTO 690
1240 AA(J,1)=22
1250 GOSUB 1280
1260 GOTO 690
1270 REM **SOUND EFFECTS
1280 CALL HCHAR(LK,LL,32)
1290 CALL SOUND(100,-7,0)
1300 CALL SOUND(100,-7,5)
1310 REM **SCORE
1320 SCORE=SCORE+23-LK
1330 FOR BB=1 TO LEN(STR$(SCORE))
1340 CALL HCHAR(1,BB+11,ASC(SEG$(STR$
     (SCORE),BB,1)))
1350 NEXT BB
1360 RETURN
```

```
1370 REM **PLAY ANOTHER GAME?
1380 TA=2
1390 X$="ANOTHER GAME?"
1400 FOR BB=1 TO LEN(X$)
1410 CALL HCHAR(24,BB+3,ASC(SEG$(X$,BB,1)))
1420 NEXT BB
1430 CALL KEY(0,K,S)
1440 IF K=89 THEN 1470
1450 IF K=78 THEN 1460 ELSE 1430
1460 END
1470 CALL HCHAR(24,3,112,15)
1480 FOR J=1 TO 7
1490 RE=AA(J,1)
1500 CE=AA(J,2)
1510 CALL HCHAR(4,1,112,30)
1520 CALL HCHAR(RE,CE,32)
1530 CALL HCHAR(1,12,32,LEN(STR$(SCORE)))
1540 SCORE=0
1550 NEXT J
1560 XX=0
1570 RESTORE 660
1580 GOTO 120
```

Variables Used in this Game Include:

LL=The beginning column of the hero
TA=Whether the game has been played
 J=Rows for upper rocks
 P=Beginning row for drawing pit

    U=Beginning column for drawing pit
    C1=Number of blocks drawn in the pit
    X$=Statements used in the game
    BB=Column for drawing statements
    XX=Whether initial positions have been
        assigned
    K=A key that has been pressed
    M=A selection of a Pitdemon
    A=Row where Pitdemon is drawn
    B=Column where Pitdemon is drawn
    AC=Changing row of Pitdemon
    AB=Number of rows a Pitdemon climbs
    AC=Present row of Pitdemon
    LK=Row of Positron Blast

Colors used in this game are as follows:
Jacque—light blue; the Pitdemons—dark red; the
Pit—yellow; the Positron Blast—white.

Movement is controlled by "S" (left) or "D"
(right). The Positron Blast is controlled by the space
bar.

Explanation of the Program

The screen clears (10) and changes to a
light yellow (20). The name of the game (30) and
directions (40-60) are displayed until the player
presses any key (70-80).

The screen clears (100), the computer
randomly selects numbers (110), and the
starting column for Jacque is set (120). A test
statement (130) allows the player to by-pass

lines 140-570. This speeds up the time between plays. The screen is then turned black (140).

Next Jacque (160, 170), the Pitdemons (180, 190), the Positron Blast (200), and a space (210) are designed.

Colors are given to the characters: Jacque (360), the Pitdemons (320), the Positron Blast (330), the space (370), and letters and numbers (230-310, 350).

The rock chamber is drawn. Line 390 uses a "dummy" variable called "J." "J" is the row number for the top of the pit.

The pit is read (430) from data lines (480-520) and drawn (460). The test (440) tells the computer that the pit is completed. The pit is drawn at row "P" (430) and Column "C" (450) "C1" (450) number of times.

The scoreboard is drawn in the first row, starting at column 3 (530-570). The actual score will be drawn later.

Jacque is drawn in his starting position (580), and the Pitdemons are assigned positions (590-660). A test (670) either draws the Pitdemons in their starting positions or allows the player to make decisions (690-730).

While the action is continuing on the screen, the computer waits for the player to make a decision. The player may move left (770-810), right (820-870), blast with the

Positron Blaster(1120-1260), or do nothing (740).

If Jacque moves left (720), he is blanked out (770) and drawn one space to the left (780-800). The test (790) insures that he won't run off the screen. After that, the routine for movement to the right (820-870) is by-passed.

If Jacque moves right (730), the computer follows the same procedure as it did for movement left. A Pitdemon is moved (860), and more decisions need to be made (690).

A test (890) either sets up initial positions or performs movement. If this is the beginning of a game, the Pitdemons are given starting positions (870-920) and are drawn (950-1080). A test (1070) checks "XX" to see if it is equal to 1. If it is, decisions need to be made again (1100).

If XX=1 (890), this means all Pitdemons have been assigned starting positions. In that case, a number if randomly chosen between 1 and 7 (900), a number for the row (910), and a number for the column (920) are chosen, and the Pitdemon is blanked out (930). A check is made (940) either to set up starting positions or perform movement. Since all Pitdemons have their starting positions, a row number is assigned (960), and a number between 1 and 3 (970) is chosen and subtracted from the row

number (980). This will cause the Pitdemon to be printed 1 to 3 rows closer to Jacque. A column number is assigned (920) and checked to see if it has moved passed Jacque (1000). If it has, it is drawn (1010), sound effects are heard (1050), and the Pitdemon's row number is changed (1060).

If the player presses the space bar (710), a blast is fired. The blast moves from Jacque to the Pitdemon (1160-1210). The blast is drawn (1170) and blanked out (1180). A check (1200) is made to see if there was a hit. If the blast misses, it continues to move (1210) until it runs out of energy. If it does, the Pitdemons move again (1220), and decisions are needed (1260).

If a Pitdemon has been hit, both the Pitdemon and the blast are blanked out (1280), and sound effects are played (1290-1300).

The score is increased by the row number minus Jacque's row number (1320). The score is then drawn (1330-1350), and more decisions are needed (1360).

If the Pitdemon reaches the rim of the pit (1000), a counter is set to signify that the game has been played one time (1380), a message is drawn (1390-1420), and the computer waits for the player to answer (1430-1450). If the player decides to play again, Jacque and the Pitdemon that reached the rim are blanked out (1470), the

other Pitdemons are blanked out (1480-1550), the score is blanked out (1510), the variable that controls whether the Pitdemons are assigned starting positions or not is set to 0 (1560), the data in line 580 can be used again (1570), and the game begins again (1580).

Some Changes You Might Like to Try:

1. Let a Warbird attack from above while the Pitdemons are climbing up from below.
2. Add some different types of creatures climbing up and give different points for them.
3. Let some of the Pitdemons go invisible and give extra points for hitting them.

In the Caverns of Carnage

Jacque and Erin are in the fearsome and eerie Caverns of Carnage, attacked by the savage Crusher beasts. How long can Jacque and Erin survive the chase that no one has ever gotten through? How far can you get them through the Caverns?

In this game, Jacque is on the right and Erin on the left. You can use Jacque's Positron Blaster to destroy the loathsome Crushers and score points for each one disintegrated. You get more points the closer they are when you blast them. But watch out! If a Crusher gets Jacque or Erin from the front, or they crash into a wall or Crusher, the game is over. Use the "S" key to move them left, the "D" key to move right, and the space bar to fire.

GAME PROGRAM LISTING

```
10 CALL CLEAR
20 CALL SCREEN(11)
```

```
 30  PRINT "CAVERNS OF CARNAGE"
 40  PRINT :::"PRESS S (LEFT) OR D (RIGHT)
     TO MOVE"
 50  PRINT :::"PRESS THE SPACE BAR TO FIRE."
 60  PRINT :::"PRESS ANY KEY TO BEGIN."
 70  CALL KEY(0,K,S)
 80  IF K<0 THEN 70
 90  CALL CLEAR
100  RANDOMIZE
110  SCORE=0
120  LL=16
130  RR=4
140  L=16
150  SW=7
160  CALL SCREEN(2)
170  REM **CHAR DESIGN
180  CALL CHAR(152,"1819FF3D3C3C2770")
190  CALL CHAR(144,"1819FFBC3C3C4280")
200  CALL CHAR(126,"183D7EBC3CCD4208")
210  CALL CHAR(33,"FFFFFFFFFFFFFFFF")
220  CALL CHAR(87,"1818181818181818")
230  CALL CHAR(96,"FFFFFFFFFFFFFFFF")
240  CALL CHAR(101,A$)
250  CALL COLOR(10,2,2)
260  CALL COLOR(2,4,4)
270  CALL COLOR(9,7,7)
280  CALL COLOR(5,8,2)
```

```
290 CALL COLOR(7,2,7)
300 CALL COLOR(11,2,2)
310 CALL COLOR(12,16,7)
320 CALL COLOR(15,8,7)
330 CALL COLOR(13,11,4)
340 CALL COLOR(6,15,2)
350 CALL COLOR(3,16,2)
360 CALL COLOR(4,16,2)
370 CALL COLOR(14,16,4)
380 CALL COLOR(16,6,7)
390 FOR JK=24 TO 1 STEP -1
400 CALL HCHAR(JK,5,96,25)
410 NEXT JK
420 CALL HCHAR(RR,LL,152)
430 CALL HCHAR(RR,LL-1,144)
440 LET RN=RND
450 IF RN>.3 THEN 480
460 L=L+1
470 GOTO 500
480 IF RN<.7 THEN 500
490 L=L-1
500 IF L<=15 THEN 520
510 L=15
520 IF L>=1 THEN 540
530 L=1
540 CALL HCHAR(24,1,33,L)
550 CALL HCHAR(24,L+1,96,SW)
```

```
560 CALL HCHAR(24,L+SW,33,32-L-SW)
570 CALL HCHAR(24,((SW-1)*RND+L+1),126)
580 CALL SOUND(100,1000*RND+110,L)
590 CALL HCHAR(RR,LL-1,96,2)
600 PRINT
610 CALL HCHAR(RR,LL,152)
620 CALL HCHAR(RR,LL-1,144)
630 CALL GCHAR(5,LL,N)
640 IF N=33 THEN 1120
650 IF N=126 THEN 1120
660 CALL GCHAR(5,LL-1,N)
670 IF N=126 THEN 1120
680 IF N=33 THEN 1120
690 CALL KEY(0,K,S)
700 IF K=32 THEN 940
710 IF K=83 THEN 760
720 IF K=68 THEN 850
730 CALL HCHAR(RR-1,LL-1,96,2)
740 GOTO 420
750 REM **INITIAL POSITIONS
760 CALL GCHAR(RR,LL-1,N)
770 IF N=126 THEN 1120
780 IF N=33 THEN 1120
790 CALL HCHAR(RR,LL-1,96,2)
800 LL=LL-1
810 IF LL<1 THEN 890
820 CALL HCHAR(RR,LL,153)
```

```
 830  CALL HCHAR(RR,LL-1,144)
 840  GOTO 1110
 850  CALL GCHAR(RR,LL+1,N)
 860  IF N=126 THEN 1120
 870  IF N=33 THEN 1120
 880  CALL HCHAR(RR,LL-1,96,2)
 890  LL=LL+1
 900  IF LL>32 THEN 800
 910  CALL HCHAR(RR,LL,153)
 920  CALL HCHAR(RR,LL-1,144)
 930  GOTO 420
 940  REM **POSITRON BLAST
 950  FOR LK=4 TO 22
 960  CALL GCHAR(LK+1,LL,N)
 970  IF N=126 THEN 1030
 980  IF N=33 THEN 420
 990  CALL HCHAR(LK+1,LL,87)
1000  CALL HCHAR(LK+1,LL,96)
1010  NEXT LK
1020  GOTO 420
1030  CALL HCHAR(LK+1,LL,96)
1040  CALL SOUND(100,-7,0)
1050  CALL SOUND(100,-7,5)
1060  REM **SCORE
1070  SCORE=SCORE+23-LK
1080  FOR BB=1 TO LEN(STR$(SCORE))
```

```
1090 CALL HCHAR(1,BB+16,ASC(SEG$(STR$
     (SCORE),BB,1)))
1100 NEXT BB
1110 GOTO 420
1120 FOR BB=1 TO LEN(STR$(SCORE))
1130 CALL HCHAR(1,BB+16,ASC(SEG$(STR$
     (SCORE),BB,1)))
1140 NEXT BB
1150 X$="ANOTHER GAME?"
1160 FOR BB=1 TO LEN(X$)
1170 CALL HCHAR(24,BB+3,ASC(SEG$(X$,BB,1)))
1180 NEXT BB
1190 CALL KEY(0,K,S)
1200 IF K=89 THEN 90
1210 IF K=78 THEN 1220 ELSE 1190
1220 END
```

Variables Used in this Game Include:

K=A key that has been pressed
LL=The row of Jacque and Erin
RR-1=The column of Erin
RR=The beginning column of Jacque
L=The beginning column of Jacque
SW=The curve of the path
JK=The rows of the entrance to the caverns
RN=A random number
LK=The row of the Positron Blast
N=A character being searched for
BB=The beginning column for a statement
X$=Statements used in the game

The colors used in this game are as follows: Jacque—dark blue; Erin—light blue; the Creatures—white; the Path—orange; the Positron Blast—black; the Walls—black.

Movement is controlled by "S" (left) or "D" (right). The space bar controls the Positron Blast.

Explanation of the Program

The screen clears (10) and changes to a light yellow (20). The name of the game (30) and directions (40-60) are displayed until the player presses any key (70-80).

The screen clears (90), the computer randomly selects numbers (100). Several variables are set: the score (110), the starting column for Jacque (120), the starting row for Jacque (130), one that allows the Crushers to appear (140), and the curve of the cavern (150).

The screen becomes black (160). Next Jacque (180), Erin (190), the Crushers (200), the walls (210), the Positron Blast (220), and a space are designed (230).

Colors are given to Jacque (380), Erin (320), the blast (290), the Crushers (310), the space (270), and letters and numbers (250, 260, 280, 300, 330, 340, 350, 360, 370).

The tunnel entrance is drawn (390-410).

Jacque (420) and Erin (430) are drawn.

The tunnel's curves are calculated (440-530) and drawn (540-570). A sound is played

(580) as Jacque and Erin are blanked out (590) and the tunnel moves up (600).

Jacque and Erin are drawn (610, 620), and a check (630 or 660) is made to see if either of them touched a wall (640 or 680) or a Crusher (650 or 670). If they touch, the game is over (1120).

While the action continues on the screen, the computer waits for the player to move left (710), right (720), or shoot (700).

If the player does nothing, Jacque and Erin are blanked out (730) and the tunnel and Crushers move closer (740).

If the player presses "S" (710), a check is made to see if Erin touched a wall (780) or a Crusher (770). If she did, the game ends (1120). If not, Jacque and Erin are blanked out (790), the column number is reduced by 1 (800) and checked (810) so that they won't run off the screen, and Jacque and Erin are drawn in their new position (820-830). The tunnel and Crushers move (840).

If the player presses "D" (720), the same procedure is followed, except Jacque's position is checked (850-870), and the column number is increased by 1 (890).

If the player presses the space bar (700), a check is made to see if a piece of the wall (980) or Crusher (970) is in the place below. If a piece

of the wall is found, the tunnel and Crushers move up (420). If the blast runs out of energy, the tunnel and Crushers move up (1030).

If a Crusher is found (970), the Crusher is blanked out (1030), sound is played (1040-1050), and the score is increased by the row number of the Crusher (1070) and drawn (1080-1100). The tunnel and Crusher move up (420).

If Jacque or Erin touch the wall or a Crusher, the game is over. The present score is drawn (1120-1140), a message is drawn (1150-1180), and the computer waits for the player either to play again (1200) or quit (1210). If the player decides to play again (1200), the game starts again. If not (1210), the game ends.

Some Changes You Might Like to Try:

1. Make other types of creatures appear and award different points.
2. Allow trap doors in the cavern as an extra challenge to Jacque and Erin.
3. Allow safe side passages as an extra aid to Jacque and Erin.

Escape

You are in the horrible prison of the Warlord and are attempting to escape with some of your fellow prisoners. But to escape, you must leave through the River of Death by passing through the Gates of Destruction.

In this game, you can help 100 prisoners to escape by guiding them through the gates. If you give no help, none will get through. No one has ever got all 100 safely through. You guide each prisoner's boat down the river and through the gates, using the "S" key for left and the "D" key for right. Because the current is so swift, there's no way you can stop the boats. You must maneuver them just right to pass through the gates—if possible.

Watch out for the rocks at the edge of the river. They will also make you lose a prisoner if one touches them. You'll also notice that the shoreline changes as the game progresses.

This represents the tide exposing and covering some of the rocks at the edge. Good luck!

GAME PROGRAM LISTING

```
 10 CALL CLEAR
 20 CALL SCREEN(11)
 30 PRINT :::"ESCAPE"
 40 PRINT :::"PRESS S (LEFT) OR D (RIGHT)
    TO STEER THE BOAT."
 50 PRINT :::"PRESS ANY KEY TO BEGIN."
 60 CALL KEY(0,K,S)
 70 IF K<0 THEN 60
 80 REM **ESCAPE
 90 NP=100
100 RANDOMIZE
110 CALL CLEAR
120 CALL SCREEN(5)
130 REM **DIM FOR BANK CHANGE
140 DIM SP(14,2)
150 FOR J=1 TO 14
160 FOR I=1 TO 2
170 READ SP(J,I)
180 NEXT I
190 NEXT J
200 DATA 6,10,6,18,8,8,8,18,10,10,10,17,12,10,12,
    18,14,10,14,18,16,8,16,14
210 DATA 18,8,18,15
220 REM **CHAR DESIGN
230 CALL CHAR(144,"FFFFFFFFFFFFFFFF")
240 CALL CHAR(101,"183C3C7E7E3C3C18")
250 REM **ASSIGN COLOR
260 CALL COLOR(15,2,2)
```

```
270 CALL COLOR(9,16,5)
280 CALL COLOR(16,13,13)
290 REM **THE BANKS
300 READ B
310 IF B=0 THEN 390
320 READ C,D
330 CALL HCHAR(B,C,152,D)
340 GOTO 300
350 DATA 5,1,10,5,17,16,6,1,10,6,18,15,7,1,9,7,19,
    16,8,1,8,8,18,15
360 DATA 9,1,9,9,17,16,10,1,10,10,17,16
370 DATA 11,1,11,11,17,16,12,1,10,12,18,15,13,1,
    12,13,16,17,14,1,10,14,18,15
380 DATA 15,1,9,15,17,16,16,1,8,16,15,19,17,1,8,17,
    13,20,18,1,8,18,15,18,19,1,9,19,16,17,0
390 REM **STATEMENTS
400 X$="ESCAPED ="
410 RR=3
420 CC=2
430 GOSUB 1730
440 X$="REMAINING ="
450 RR=3
460 CC=16
470 GOSUB 1730
480 NP=NP-1
490 X=NP
500 CC=28
510 GOSUB 1770
520 GOTO 1040
```

```
530 NP=NP-1
540 IF NP<0 THEN 590
550 X=NP
560 RR=3
570 CC=28
580 GOSUB 1770
590 PE=PE+1
600 X=PE
610 RR=3
620 CC=12
630 GOSUB 1770
640 X$="FREEDOM!!!!!!"
650 CALL HCHAR(A,B,152,2)
660 RR=4
670 CC=7
680 GOSUB 1730
690 CALL SOUND(150,523,3)
700 CALL SOUND(80,523,3)
710 CALL SOUND(150,523,3)
720 CALL SOUND(300,700,3)
730 GOTO 990
740 CALL GCHAR(DI+1,LL,OLDCHAR)
745 CALL HCHAR(DI+1,LL,101)
750 CALL SOUND(100,-7,0)
760 CALL SOUND(100,-7,5)
770 FOR J=1 TO 5
780 CALL COLOR(9,7,5)
790 FOR I=1 TO 10
800 NEXT I
```

```
 810 CALL COLOR(9,16,5)
 820 NEXT J
 830 CALL HCHAR(DI+1,LL,OLDCHAR)
 840 NP=NP-1
 850 IF NP<0 THEN 900
 860 X=NP
 870 RR=3
 880 CC=28
 890 GOSUB 1770
 900 X$="ANOTHER LOST!!"
 910 CALL HCHAR(A,B,152,2)
 920 RR=4
 930 CC=7
 940 GOSUB 1730
 950 CALL SOUND(200,262,2)
 960 CALL SOUND(600,131,0)
 970 FOR J=1 TO 100
 980 NEXT J
 990 CALL HCHAR(4,CC+1,32,LEN(X$))
1000 IF NP<0 THEN 1820
1010 LK=23
1020 LL=13
1025 IF FLAG=1 THEN 1100
1030 GOTO 1480
1040 LK=24
1050 LL=13
1060 ROW=5
1070 M=M+1
1080 CALL GCHAR(ROW,M,N)
```

```
1082 IF N=101 THEN 1084 ELSE 1090
1084 OLDCHAR=32
1085 FLAG=1
1086 DI=LK-1
1088 GOTO 750
1090 IF N=32 THEN 1100 ELSE 1070
1100 C=M-1
1105 FLAG=0
1110 C=C+1
1120 SR=INT(RND*13)+1
1130 A=SP(SR,1)
1140 B=SP(SR,2)
1150 CALL HCHAR(A,B,32,2)
1160 IF C=21 THEN 1500
1170 CALL GCHAR(ROW,C,N)
1180 T=T+1
1190 IF N=101 THEN 1192 ELSE 1200
1192 OLDCHAR=144
1194 DI=LK-1
1196 GOTO 750
1200 IF N=152 THEN 1500
1210 CALL HCHAR(ROW,C,144)
1220 CALL SOUND(100,20*C,20-C)
1270 CALL KEY(0,K,S)
1275 CALL HCHAR(LK,LL,32)
1280 IF K=83 THEN 1370
1290 IF K=68 THEN 1370
1330 LK=LK-1
1340 GOSUB 1620
```

```
1350 GOTO 1480
1360 REM **HERO MOVE
1370 GOSUB 1620
1380 CALL HCHAR(LK,LL,32)
1390 IF K=83 THEN 1440
1400 LL=LL+1
1410 LK=LK-1
1420 GOSUB 1620
1430 GOTO 1480
1440 LL=LL-1
1450 LK=LK-1
1460 GOSUB 1620
1480 CALL HCHAR(LK,LL,101)
1490 GOTO 1110
1500 W=W+1
1510 CALL HCHAR(ROW,M,32,T-1)
1520 T=0
1530 IF W>7 THEN 1570
1540 ROW=ROW+2
1550 M=0
1560 GOTO 1070
1570 W=0
1590 M=0
1600 ROW=0
1610 GOTO 1060
1620 CALL GCHAR(LK-1,LL,N)
1630 DI=LK-1
1640 IF N=144 THEN 740
1650 IF N=152 THEN 740
```

```
1660 CALL GCHAR(LK,LL,N)
1670 DI=LK
1680 IF N=144 THEN 740
1690 IF N=152 THEN 740
1700 IF LK<5 THEN 1710 ELSE 1720
1710 GOSUB 530
1720 RETURN
1730 FOR BB=1 TO LEN(X$)
1740 CALL HCHAR(RR,BB+CC,ASC(SEG$(X$,BB,1)))
1750 NEXT BB
1760 RETURN
1770 FOR BB=1 TO LEN(STR$(X))
1780 CALL HCHAR(RR,BB+CC,ASC(SEG$(STR$(X),
     BB,1)))
1790 NEXT BB
1800 CALL HCHAR(RR,BB+CC,32)
1810 RETURN
1820 REM **PLAY ANOTHER GAME
1830 TA=2
1840 X$="ANOTHER GAME?"
1850 FOR BB=1 TO LEN(X$)
1860 CALL HCHAR(24,BB+10,ASC(SEG$(X$,BB,1)))
1870 NEXT BB
1880 CALL KEY(0,K,S)
1890 RESTORE
1900 PE=0
1910 C=0
1920 W=0
1930 M=0
```

```
1940 NP=100
1950 IF K=89 THEN 90
1960 IF K=78 THEN 1980 ELSE 1880
1970 GOTO 90
1980 END
```

Variables Used in this Game Include:

K=A key that has been pressed
B=The row for a river bank
C=The column for a river bank
D=The length of a river bank
RR=The row where a statement is drawn
CC=The beginning column of a statement
X$=The statement
NP=The number of prisoners
PE=The number of prisoners that
    have escaped
X=Either NP or PE
A=The row of a bank change
B=The column of a bank change
DI=The destroyed boat
LL=The beginning column of a boat
LK=The row of a boat
ROW=The starting row of the gate
M=The beginning and ending of a gate
N=The character being searched for
T=The length of the gate
W=The number of gates that have
    been drawn
BB=The column for a statement
TA=Whether the game has been played

OLDCHAR=The original character which will be
            replaced after flashing to show a
            collision
   FLAG=Whether the gate has smashed into the
            boat on the first move of the gate away
            from the shore (FLAG=1)

Colors used in this game are: the Gate—black;
the Boat—white; the River Banks—green; and the
Water—blue.

Movement is controlled by "S" (left) or "D"
(right).

Explanation of the Program

The screen clears (10) and changes to a
light yellow (20), and the directions are drawn
(30-40). The computer waits for the player to
press any key (60, 70).

The screen clears (110), the computer is
told to select numbers randomly (100), the
number of prisoners is set (90), and the screen
turns blue (120).

A dimension (140) sets aside memory for
the bank's changing shape. Information is read
(150-190) from lines 200 and 210. The
information is then stored in the dimension
statement until it is needed. For example, the
first number in the statement "SP (6, 10)" is the
row number, and the second number is the
column number.

The gate (230) and the boat (240) are
designed.

Color is given to the gate (260), the boat (270), and the banks (280).

Information is read (300) from data lines (350-380). The information is first checked for a zero (310)—this tells the computer the banks are completed, then more information is read (320). The gate is drawn (330), and the procedure is repeated until a zero is found.

Next a variable is assigned a string (400), a row number (410), and a column number (420) and is drawn (430, 1730-1760). This procedure is repeated again for lines 690-720, and decisions need to be made. (520)

This section is used when a prisoner escapes (530-730) or is lost (740-1000). Basically, a phrase is assigned (640 or 900) and drawn (660-680 or 920-940), music is played (690-720 or 950, 960), the boat either vanishes or explodes (740-820), and the score is adjusted (530-630 or 840-890).

These lines are used after the game has started. The row number (1010) and column number (1020) are assigned, and the movement procedure is continued (1030).

The boat's starting row (1040) and column (1050) are assigned, and the gate's starting row is assigned (1060).

The gate's starting column is determined by (1070-1090). The computer checks each

column in the current row. As soon as a space
is found (1040), the column is identified (1100).
The banks are randomly changed by (1120-
1150), while (1082-1088) check if the first gate
movement hit the boat.

As the gate is drawn (1100-1190, 1620-
1720) rows and columns are checked for either
a boat (1190) or a bank (1200). At the same time,
the boat checks for a gate or a bank (1620-
1720). If the checks fail, sounds are played
(1220) and the boat and gate continue to move.

The player has a choice of moving left
(1280) or right (1290). Moving left or right
causes checks to be made for banks or gates
(1620-1720). If a gate or bank is found, the boat
explodes (740-820), the score is adjusted and
drawn (840-890), and the appropriate message
is shown (900-1000).

If nothing is found, the boat either moves
left (1360, 1440-1490) or right (1400-1430,
1480-1490).

Each time a gate meets the next bank, a
counter is increased by 1 (1500), and the gate is
blanked out (1510) and checked to see if 7
gates have been drawn (1530). If there have
been 7 gates, counters are reset (1570-1600),
and the procedure above is repeated. If not, a
counter is reset that controls the starting
position of the gate (1550) to zero, and the row
number is increased by two (1540).

When all prisoners have either escaped or lost (540 or 850), the game ends. A variable is reset (1520) to indicate that the game has been played one time. A question is drawn (1810-1860), and the computer waits for the player to answer (1870-1950). If the player chooses, the game is restarted (1960).

Some Changes You Might Like to Try:

1. Allow multiple gates to make it harder for the prisoners to get through.
2. Allow the prisoners boats to stop temporarily.
3. Make rocks randomly appear in the water to make it harder for the boats to get through.
4. Make the shoreline follow a sine curve.

Rockfall!

Things are bad, bad, bad. In fact, they're worse. The Wizard's creatures are attacking, and rocks are crashing down as Erin pulls the vine. How long can Jacque and Erin survive?

In this game, Jacque is on the right, and Erin is on the left. The Wizard's creatures attack from the sides. The following controls are used:

"S" moves Jacque and Erin left
"D" moves Jacque and Erin right
"<" fires the Positron Blaster left
Space bar fires the blaster up
">" fires the blaster right

If a rock hits Jacque or Erin, the game is over. Likewise, if a creature gets to either of them, the game ends.

The rocks initially start falling from near the middle of the screen. When the rocks reach the row just above the people, the rocks will disappear and a new row will appear if the game isn't over.

This new row will start one row closer to the heroes. Now you'll get one more point per rock disintegrated. The rocks will keep starting off one row closer until row 15, and you'll get 15

points per rock or creature destroyed at this time.

Creatures can appear, starting from two spaces to the right or left of Jacque. As soon as you disintegrate a creature, another will appear at a different location and will start moving toward Jacque and Erin. You score more points the closer the creatures get before you blast them. Your score will be continually updated on the screen.

GAME PROGRAM LISTING

```
 10 CALL CLEAR
 20 CALL SCREEN(11)
 30 PRINT "ROCKFALL"
 40 PRINT :::"PRESS S (LEFT) OR D (RIGHT)
    TO MOVE."
 50 PRINT :::"PRESS < (LEFT) OR > (RIGHT) TO FIRE
    HORIZONTALLY."
 60 PRINT :::"PRESS THE SPACE BAR TO FIRE
    VERTICALLY."
 70 PRINT :::"PRESS ANY KEY TO BEGIN."
 80 CALL KEY(0,K,S)
 90 IF K<0 THEN 80
100 CALL CLEAR
110 RANDOMIZE
120 NR=15
130 DIM A(21)
140 SCORE=0
150 G=25
```

```
160 SR=11
170 R=11
180 F=10
190 LL=16
195 NB=5
200 CALL SCREEN(12)
210 REM **CHAR DESIGN
220 CALL CHAR(152,"1819FF3D3C3C2770")
230 CALL CHAR(153,"1819FFBC3C3C4280")
240 CALL CHAR(144,"1819FFBC3C3C2720")
250 CALL CHAR(145,"1898FF3D3C3C2770")
260 CALL CHAR(136,"18387C75737C2442")
270 CALL CHAR(137,"18387D73707C2418")
280 CALL CHAR(138,"181C3757C7372452")
290 CALL CHAR(139,"181CBECE1E3E2418")
300 CALL CHAR(88,"000000FFFF")
310 CALL CHAR(126,"FFFF7E7E3C3C3C18")
320 CALL CHAR(89,"0")
330 CALL CHAR(87,"1818181818181818")
340 CALL CHAR(96,"FFFFFFFFFFFFFFFF")
350 REM **ASSIGN COLOR
360 CALL COLOR(2,4,4)
370 CALL COLOR(9,2,2)
380 CALL COLOR(11,2,2)
390 CALL COLOR(12,2,12)
400 CALL COLOR(13,11,4)
410 CALL COLOR(14,16,12)
420 CALL COLOR(15,7,12)
```

```
430  CALL COLOR(16,6,12)
440  REM **ROCKS
450  FOR J=1 TO 10
460  CALL HCHAR(J,1,112,32)
470  NEXT J
480  REM **SCOREBOARD
490  X$="SCORE = "
500  FOR BB=1 TO LEN(X$)
510  CALL HCHAR(9,BB+10,ASC(SEG$(X$,BB,1)))
520  NEXT BB
530  FOR I=1 TO NR
540  A(I)=126
550  NEXT I
560  FOR I=1 TO NB
570  A(NR*RND)=32
580  NEXT I
590  R=SR
600  CALL HCHAR(20,LL,152)
610  CALL HCHAR(20,LL-1,144)
620  REM **DECISIONS
630  IF R=20 THEN 530
640  GOSUB 1590
650  CALL KEY(0,K,S)
660  IF K=32 THEN 960
670  IF K=83 THEN 720
680  IF K=68 THEN 790
690  IF K=44 THEN 1270
700  IF K=46 THEN 1390
```

```
710 GOTO 630
720 CALL HCHAR(20,LL,32)
730 CALL HCHAR(20,LL-1,32)
740 LL=LL-1
750 IF LL<12 THEN 810
760 CALL HCHAR(20,LL,153)
770 CALL HCHAR(20,LL-1,145)
780 GOTO 850
790 CALL HCHAR(20,LL,32)
800 CALL HCHAR(20,LL-1,32)
810 LL=LL+1
820 IF LL>21 THEN 740
830 CALL HCHAR(20,LL,153)
840 CALL HCHAR(20,LL-1,145)
850 FF=INT(RND*2)+1
860 CALL HCHAR(20,F,32)
870 F=F+FF
880 IF F>=LL-2 THEN 1130
890 CALL HCHAR(20,F,136)
900 GG=INT(RND*2)+1
910 CALL HCHAR(20,G,32)
920 G=G-GG
930 IF G<=LL THEN 1150
940 CALL HCHAR(20,G,139)
950 GOTO 630
960 FOR LK=19 TO R STEP -1
970 CALL HCHAR(LK,LL,87)
980 CALL HCHAR(LK,LL,32)
```

```
 990 NEXT LK
1000 IF A(LL-10)<>32 THEN 1020
1010 GOTO 630
1020 A(LL-10)=32
1030 SCORE=SCORE+R
1040 GOSUB 1060
1050 GOTO 630
1060 CALL SOUND(100,-7,0)
1070 CALL SOUND(100,-7,5)
1080 REM **SCORE
1090 FOR BB=1 TO LEN(STR$(SCORE))
1100 CALL HCHAR(9,BB+18,ASC(SEG$(STR$
     (SCORE),BB,1)))
1110 NEXT BB
1120 RETURN
1130 CALL HCHAR(20,LL-1,137)
1140 GOTO 1170
1150 CALL HCHAR(20,LL,138)
1160 REM **TRY AGAIN
1170 X$="ANOTHER GAME?"
1180 FOR BB=1 TO LEN(X$)
1190 CALL HCHAR(24,BB+3,ASC(SEG$(X$,BB,1)))
1200 NEXT BB
1210 RESTORE
1220 CALL KEY(O,K,S)
1230 IF K=89 THEN 100
1240 IF K=78 THEN 1250 ELSE 1220
1250 END
```

```
1260 REM **HORIZONTAL BLAST
1270 SH=LL-2
1280 FOR TH=SH TO 11 STEP -1
1290 CALL HCHAR(20,TH,88)
1300 CALL HCHAR(20,TH,32)
1310 IF TH=F THEN 1340
1320 NEXT TH
1330 GOTO 630
1340 SCORE=SCORE+17-LL+F
1350 F=10
1360 CALL HCHAR(20,TH,32)
1370 GOSUB 1060
1380 GOTO 630
1390 FOR OH=LL+1 TO 24
1400 CALL HCHAR(20,OH,88)
1410 CALL HCHAR(20,OH,32)
1420 TH=OH
1430 IF OH=G THEN 1460
1440 NEXT OH
1450 GOTO 630
1460 SCORE=SCORE+17+LL-G
1470 G=25
1480 CALL HCHAR(20,OH,32)
1490 GOTO 1370
1500 IF AA(M,2)=LL THEN 1170
1510 IF AA(M,2)=LL-1 THEN 1170
1520 CALL HCHAR(AC,B,32)
1530 AA(M,1)=11
```

```
1540 RETURN
1550 CALL SOUND(100,-7,0)
1560 CALL SOUND(100,-7,5)
1570 RETURN
1580 REM **ROCK FALL
1590 R=R+1
1600 IF R<20 THEN 1690
1610 CALL GCHAR(19,LL,N)
1620 IF N=126 THEN 1170
1630 CALL GCHAR(19,LL-1,N)
1640 IF N=126 THEN 1170
1650 IF SR<15 THEN 1660 ELSE 1670
1660 SR=SR+1
1670 CALL HCHAR(R-1,10,32,17)
1680 RETURN
1690 CALL HCHAR(R-1,10,32,17)
1700 FOR JJ=1 TO 15
1710 CALL HCHAR(R,10+JJ,A(JJ))
1720 NEXT JJ
1730 CALL SOUND(100,10*R,20-R)
1740 IF F>10 THEN 1790
1750 F=11+INT(2*RND)
1760 IF F>LL-2 THEN 1750
1770 CALL HCHAR(20,F,136)
1780 RETURN
1790 IF G<25 THEN 1830
1800 G=24-INT(2*RND)
1810 IF G<LL+2 THEN 1800
```

1820 CALL HCHAR(20,G,138)
1830 RETURN

Variables Used in this Game Include:

K=A key that has been pressed
LL=The row of Jacque
RR=The beginning column of Jacque
L=The width of the cavern path
SW=The curve of the cavern path
JK=A row of entrance to the cavern
RN=A random number
LK=Row of Positron Blast
N=Character being searched for
BB=Beginning column for statements
X$=Statements used in the game

The colors used in this game are as follows:
Jacque—dark blue; Erin—light blue; the Creatures—
dark red; the Path—orange; the Positron Blast—
black; Spaces—orange; the Walls—black.

Movement is controlled by "S" (left) or "D"
(right). The space bar controls the vertical Positron
Blast, and "<" or ">" control the horizontal Positron
Blast.

Explanation of the Program

The screen clears (10) and changes to a
light yellow (20). The name of the game (30) and
directions (40-70) are displayed until the player
presses any key (80-90).

The screen clears (100), and the computer
randomly selects numbers (110). Several
variables are set: the number of rocks (120), a

dimension of possible characters in a row (130), the score (140), the starting column of the Crusher that moves left (150), the beginning row for the rockfall (160-170), the starting column number of the Crusher that moves right (180), and the starting column of Jacque (190). The screen is then turned light green (200).

Next Jacque (220, 230), Erin (240, 250), the right Crusher (260, 270), the left Crusher (280, 290), the vertical Positron Blast (300), the rock (310), a space (320), the horizontal Positron Blast (330), and the pit wall (340) are designed.

Colors are given to the characters: Jacque (430), the Crushers (410), Erin (420), the rocks (390), the wall (370), and spaces (360, 380, 400).

The upper rock chamber is drawn (440-470).

The scoreboard is drawn in the ninth row and starting in the tenth column (490-520). The actual score will be drawn later.

First, the rocks are set to the ASCII value of 126 (530-550). Next, five rocks are randomly selected and assigned the ASCII value of 32 (560-580). These five rocks are inserted randomly inside of the rockfall. This means that each new rockfall will be different.

Jacque and Erin are drawn in their starting positions (600-610).

While the action continues on the screen, the computer waits for the player to make a

decision. The player may move left (720-780), right (790-840), blast the Positron up (960-1010) or across (1260-1490), or do nothing (710).

If Jacque and Erin move left (670), they're blanked out (720, 730) and drawn one space to the left (760-770). The test (750) insures they'll not run off the screen. After that, the program by-passes the routine for movement to the right and Crushers are moved (870-950) and more decisions need to be made.

If Jacque and Erin move right (670), the computer follows the same procedure as it did for movement left.

If no decisions are made (710), the row number of the rockfall is checked (630). If it is equal to 20, then a new rockfall is designed (530-590). If the row number does not equal 20, the row is checked (1590-1600). Locations are checked above Jacque (1610-1620) and Erin (1630-1640). If a rock is present, the game ends (1620 or 1640). If a space is present, the rockfall is blanked out (1690) and redrawn (1700-1720), and a sound is made (1730).

If the player pressed the space bar (660), a Positron Blast is shot towards the rocks (960-990). If the position of the blast hits a rock (1000), the rock is blanked out (980), the score is increased by the row number of the rock (1030), a sound is made (1060-1070), the rock's

position is replaced by a space (1020), and the score is drawn (1080-1110).

If the player chooses to shoot left (690), the blast appears one space to the left of Erin (1270). As the blast moves towards the Crusher, it is drawn and blanked out (1280-1320). When the blast's position equals the position of the Crusher (1310), the Crusher is assigned a new position (1350) and blanked out (1360), the score is increased by the column number of the Crusher minus the column row of Jacque (1460), sound is made (1060-1070) and drawn (1740-1780), and more decisions need to be made.

If the player chooses to shoot right (700), the same procedure is followed (1390-1490).

When the game is over a message is drawn (1160-1200), data can be used again (1210), and the computer waits for the player to answer. If the player presses "Y," the game begins again (1220). If the player presses "N," then the game ends (1240-1250).

Some Changes You Might Like to Try:
1. Make the creatures appear and disappear so that it is harder to hit them.
2. Allow some colored rocks or Warbirds to appear and give more points for shooting them.

Timelost is the first book in a new unique series of integrated comic adventures and computer action games.

A teenage computer whiz and his young sister are swept through Time Holes and meet attacking Warbirds of the past, a mysterious man from the future, and a sinister Time Wizard.

Each lost-in-time adventure is combined with an arcade-type computer action game. Explanations help new users to learn programming and suggested variations challenge young users to create more game fun.