

THE CONTENTS OF THIS MANUAL REVEAL MANY OF THE MYSTERIES OF THE GAME. THE GAME DIRECTOR, OR "OPERATOR" MAY WISH TO CONCEAL THE CONTENTS FROM THE PLAYERS IN ORDER TO ENHANCE THE EXCITEMENT OF THE GAME. THE OPERATOR SHOULD PLAY THE GAME BEFORE READING THE LISTINGS!

KIM—VENTURE

CASSETTE INFORMATION

Thirty seconds (30 sec.) of SYNC characters first. KIM speed.

ID	Loads	Time	
A1	000 - 0F0	:35	After loading all three segments of
A2	0100 - 03FF	1:40	the game, use the key sequence <u>AD</u> ,
A3	1780 - 17E6	:20	0100, <u>GQ</u> to start the game.
06	0100 - 0274		Scoring Program. Do not load this segment until you are ready to stop the game.

The information in this manual has been reviewed and is believed to be entirely reliable. However, no responsibility is assumed by either Robert Leedom or ARESCO, Inc. for any inaccuracies. The material in this manual is for informational purposes only and is subject to change without notice.

© 1979, Robert Leedom

ARESCO, Inc. * P O Box 1142 * Columbia * MD * 21044

KIM-VENTURE OPERATION

These notes do not relate to the play of the game, but rather to the loading and running of the program by someone familiar with the KIM-1 microcomputer. Having followed these instructions, the "operator" may then turn the game over to a player (or group of players), giving the player only the sheets labeled KIM-VENTURE INSTRUCTIONS.

LOADING THE PROGRAM

It is assumed that the operator knows the basic procedure for loading programs from audio tape. If not, the subject is well covered on page 47 of the KIM-1 User Manual. The KIM-VENTURE cassette consists of approximately 30 seconds of SYNC characters, followed by three program segments. The segments have been given ID numbers A1, A2, and A3.

To load the program, you must load the first segment (ID A1); stop the recorder while you change the ID to A2, then load the second segment. Finally, stop the recorder while changing the ID to A3, and load the last segment of the program.

RUNNING THE PROGRAM

1. Once the program has been loaded, start the game with the key sequence: AD , 0,1,0,0, GO
2. At any time, the game can be halted by pressing the RESET key (RS). The game may be resumed at the point of interruption by using the key sequence AD , 0,1,0,0, GO.
3. To start a new game without having to reload the whole program, start by loading only the first segment (ID=A1). Check location 03BD to be sure that it contains 0E. It may have been changed to 05; if so, enter 0E before going on.

The change can be done by using the key sequence AD,0,3,B,D, DA, 0,E. Then begin the game again using the key sequence specified in step 1 (AD ,0,1,0,0, GO).

4. If the players wish to stop the game and resume play at a later date, the operator can save the game status by dumping page zero (locations 0000 through 00EE) on cassette. Label the cassette STATUS DATA. Write the data stored in location 03BD on the tape label along with the title. This data will be needed when play is resumed. When ready to begin the game again (from the point at which it was stopped), load sections A1, A2, and A3, then load your stored page zero Status Data. Load location 03BD with the data written on the cassette label, then start the game as usual at location 0100.

SCORING

To check on a player's score, stop the game using the RS (reset) key. Then load the scoring program (ID=06). Remember to set F1=0! Then enter AD ,0100, GO . KIM will display the word SCORE, then the player's CLASS and the number of moves it took that player to reach that class. The object, of course, is to obtain the lowest number of moves ~~(which corresponds to the highest class)~~.

CLASS	TITLE	QUALIFICATIONS
A	Grandmaster	Deposited both treasures inside the house in 40 moves or less
B	Master	Deposited both treasures inside the house in 41 moves or more
C	Junior Master	Deposited one treasure inside the house
D	Seasoned Adventurer	Visited all areas of the caverns
E	Advanced Adventurer	Found both treasures
F	Experienced Adventurer	Got stuck in the caverns. If a player gets stuck, he cannot earn a higher rating!
G	Explorer	Passed the first big obstacle - OR - used the F key correctly
H	Amateur	Entered the caverns
I	Novice	Found three objects
J	Beginner	Explored all the above-ground locations
O	Unrated	None

7	-			3	7
Class				# of Moves	

TO CHANGE THE DISPLAY RATE

If KIM "text" is unfamiliar (for example, "~~ur~~ At " is read "You are at"), the operator may want to slow down the display to give the players more time to examine it. Changing the data in location 02EA from C0 to a larger number (for example, E0) will slow the display rate. Once the players have become familiar with the text, the operator may wish to "speed up" the game somewhat by increasing the speed of the display rate. A value smaller than C0 in location 02EA (such as 50) will speed it up. Experiment to determine which display rate is most comfortable for the players.

PROGRESS CHECK

"Losing" is doing something during the course of the adventure that makes it impossible to get any further in the game. There are no messages to this effect, since it will usually take the players several moments to come to the conclusion that they are "trapped in the caves". If the players think they've lost, the operator can check their "score" by pressing RESET (RS), and checking the following locations:

If location 0040 contains 01, the player cannot get any deeper into the caves because the necessary resources have been used up.

If location 0045 contains 06 AND if location 003E does not have bit 2 set to 1 (that is, if the contents of 003E, AND'ed with \$04 is not equal to \$04), then the player is stuck right where he is...and he doesn't have what he needs to get out.

If neither of the above conditions is true, the player is still in the game. Continue, using AD, 0100, GO.

If the player has not "lost" the game, he may still find himself unable to retrieve the treasures, and may concede a "loss". In this event, he receives no score at all. If the player retrieves only one treasure, he receives no score. In order to "win" the game, the player must retrieve both treasures and leave them in the cellar of the house. } Ignore

Many players come to the conclusion that they can leave the treasure "at" the house, rather than "in" it; the operator may or may not wish to call this error to the player's attention.

The operator may wish to provide pencils and paper to the players so they can keep track of where they have been and of the "symbols" displayed. Many players will draw a "map" of the caves as they explore, drawing in all the objects and monsters encountered. The operator should cooperate, because the players will feel more confident of their ability to "figure it out". In addition, the operator may wish to keep a running tally of the number of locations visited by each player (or group of players). This "score" can then be used to award titles such as "grand master" or "adventurer extraordinaire" to the player with the least number of moves. } DON'T PEEK! Ignore

The listing provided with the game is for the operator's use, and (as with these notes) should not be shown to the players. The operator might find it desirable to make modifications or enhancements, since this version of KIM-VENTURE is designed to be run on a "bare-bones" KIM.

We welcome any comments and/or suggestions regarding your experiences with KIM-VENTURE. Please do not hesitate to offer your remarks, since we are definitely interested in doing all we can to improve and upgrade the game.

Robert Leedom

KIM-VENTURE LISTING

```

; KIM-VENTURE      © Copyright R.C.Leedom 1979
;
; LIGHT Subroutine. Lights KIM 7-segment dis-
; plays with character-codes contained
; in table WINDO. On return, key from
; keyboard is in A-reg (else A-reg=$15).
; Y-reg is preserved.
0000 84 EF      LIGHT STY YSAV      Save Y-register.
0002 A0 00      LDY #0
0004 A9 7F      LDA #$7F      Set directional
0006 8D 41 17    STA PADD      registers.
0009 A2 09      LDX #9      Start with leftmost
000B 84 FC      LITELP STY TEMP    character.
000D B9 F0 00    LDA WINDO,Y    Get char to be shown.
0010 20 4E 1F    JSR CONVD+6    Use KIM monitor subr.
0013 C8         INY           Next char on right ...
0014 C0 06      CPY #6      Done all six yet?
0016 90 F3      BCC LITELP    Not yet. Continue.

;
0018 20 3D 1F    KBG      JSR KEYS      Before return, sample
001B 20 6A 1F    JSR GETKEY    the KIM keyboard.
001E A4 EF      LDY YSAV      Restore Y-register.
0020 60         RTS

;
; Character look-up table. Frequently-used
; characters stored as 4-bit pointers
; into part of this table (FUTBL). In-
; frequently-used characters stored as
; 8-bit pointers into the other part
; (IUTBL). Note that the last two en-
; tries are variables, used for special
; program-controllable characters.
;
; Index Character
; FUTBL 2 A
;        3 C
;        4 D
;        5 E
;        6 H
;        7 I
;        8 L
;        9 N
;        A O
;        B R
;        C S
;        D T
;        E U
;        F (blank)
; IUTBL 10 - (dash)
;        11 B
;        12 F
;        13 G
0021 77
0022 39
0023 5E
0024 79
0025 76
0026 06
0027 38
0028 54
0029 5C
002A 50
002B 6D
002C 78
002D 1C
002E 00
002F 40
0030 7C
0031 71
0032 3D

```

```

; (continuation of IUTBL)
0033 1E      14      J
0034 37      15      M
0035 73      16      P
0036 3E      17      W
0037 6E      18      Y
0038 53      19      ?
0039 08      1A      . (period)
003A 5B      1B      2
003B 00      1C      RES 1      Variable (signpost)
003C 00      1D      RES 1      Variable (? or magic
                                button)
;
;
; Program variables (except for a very few
; located elsewhere)
003D 00      NMBUTS    No. of magic button uses
003E 00      BURDEN    Bit #n set if carrying object #n.
003F 00      DLOBAD     Abs address of obj to be deleted.
0040 FF      DRAGON     FF=hungry; 0=dead; 1=full.
0041 EC      EGOLAD     Current address of EGO file.
0042 00      LOCAD      Addr of current loaction file.
0043 00      LINTAX     Pointer; 0=EGOLAD; 1=LOCAD.
0044 00      LOBJAD     Addr of last obj in file, or
                        of object of interest.
0045 0B      LOCNUM     Number of current location.
0046 FF      MBUT       Current magic button (0 - F valid).
0047 00      NOBCRY     No. of objects carried (0 - 4).
0048 00      NOBS       No. of objs to be displayed (0 - 7).
0049 00      OBJ        Object identifier (0 - 7).
004A 00      POINTR     ADL of message
004B 03      ADH         ADH of message (Constant!)
004C 00      MOVES      L.S. Half of number of moves.
;
; Program constants
;
004D 02      OBJMSK     B00000010 (1) Bird      When obj is
004E 04      B00000100 (2) Rope      picked up (or is
004F 08      B00001000 (3) Rod      dropped), the
0050 10      B00010000 (4) File      proper bit is
0051 20      B00100000 (5) Cage      Ored into (or is
0052 40      B01000000 (6) Pearls   Nanded out of)
0053 80      B10000000 (7) Gold     BURDEN.
;
; Message addresses. These are the ADL's of
; the messages, all of which are assumed
; to reside in page 3 (see POINTR+1).
; Order of this table is paramount!
; There is a variable thrown in here to
; separate ADOPGR and ADBRDG ...

```

0054 34	ADOPGR +OPNGRM	Open Grate	
0055 00	SCDU RES 1	-1,0,1,2 : Browse, Carry, Drop, Use	
0056 2B	ADBRDG +BRAGM	Bridge Across Gully	
0057 BE	OBMSAD +ADDRAG	(0) Dragon	
0058 DC	+BIRD	(1) Bird	
0059 43	+ROPE	(2) Rope	
005A E4	+ROD	(3) Rod	
005B 25	+FILE	(4) File	
005C 22	+CAGE	(5) Cage	
005D 1C	+PEARLS	(6) Pearls	
005E 46	+GOLD	(7) Gold	
005F 8C	UINMAD +URIN	You Are In	
0060 89	+URAT	You Are At	
0061 B5	+ISEE	I See -	
0062 E7	CYMSAD +CARRY	Carry -	
0063 D7	+DROP	Drop -	
0064 AE	+USE	Use -	
0065 06	LNAMAD +CELLAR	Cellar	0
0066 09	+PURPLO	Purple Oracle	1
0067 11	+REDRM	Red Room	2
0068 18	ADSSM +STSTPS	Stone Steps	3
0069 D3	+BLUDEN	Blue Den	4
006A 3C	ADGRM +STGRAT	Steel Grate	5
006B 4F	+HOLE	Hole	6
006C 38	ADGYM +GULLY	Gully	7
006D 7B	ADRHM +RYHALL	Royal Hall	8
006E AB	+HOUSE	House	9
006F DB	+BIRDRM	Bird Room	A
0070 70	+STREAM	Stream	B
0071 52	ADTSM +TSHAFT	Tight Shaft	C
0072 93	+NPIT	N. Pit	D
0073 EF	+GROTTO	Grotto	E
0074 6A	+OYSTRB	Oyster-bed	F
0075 28	+CHUTE	Chute	10
0076 73	+EPIT	E. Pit	11
0077 9B	+ATTIC	Attic	12
0078 EB	+SPIT	S. Pit	13
0079 56	+TUNNEL	Tunnel	14
007A 5D	+SLIT	2-Inch Slit	15
007B 82	+GLEN	Glen	16
007C F3	+FOREST	Forest	17

CAVE MAP

Format for file for each location in caves is as follows:

Word #	Contents
0	Bit 7 = 1, bit 6 = 0. Bit 5 = 1 if location has been visited during the game. Bits 4 - 0 contain the location number of this file.
1	Bit 7 = 0. Bit 6 = 1 if magic button works in this location. Bits 5,4,3,2,1,0 = 1 if you can leave this location in the D,U,W,S,E,N direction, respectively. This word is used as the "signpost" in the Cue message.
next (up to) six words	Bit 7 = 0, bit 6 = 1. Bits 5 - 0 specify a location to which you may move from this location. The first of these words specifies the destination for the lowest-numbered bit which is set in word 1; the second specifies the destination for the next-lowest bit set in word 1, etc. Therefore, there must be one of these words for each of the first six bits (5 - 0) set in word 1 of this file.
next (up to) eight words	Bit 7 = 0, bit 6 = 0. Bits 5 - 0 specify the object number of an object at this location. There may be as many as eight of these words, or there may be none at all.
007D 88	LOCNUM = 8 <u>Royal Hall</u>
007E 3F	Directions: N,E,S,W,U,D
007F 4E	N to Grotto (E)
0080 43	E to Stone Steps (3)
0081 4C	S to Tight Shaft (C)
0082 4D	W to N. Pit (D)
0083 52	U to Attic (12)
0084 50	D to Chute (10)
0085 00 DRAGAD	Dragon
0086 94	LOCNUM = 14 <u>Tunnel</u>
0087 11	Directions: N,U
0088 4A	N to Bird Room (A)
0089 45	U to Steel Grate (5)
008A 03	Rod

;		
008B 95	LOCNUM = 15	<u>2-Inch Slit</u>
008C 05	Directions: N,S	
008D 4B	N to Stream (B)	
008E 45	S to Steel Grate (5)	
;		
008F 96	LOCNUM = 16	<u>Glen</u>
0090 02	Directions: E	
0091 4B	E to Stream (B)	
;		
0092 8F	LOCNUM = F	<u>Oyster-Bed</u>
0093 10	Directions: U	
0094 44	U to Blue Den (4)	
0095 06	Pearls	
;		
0096 80	LOCNUM = 0	<u>Cellar</u>
0097 50	Directions: U, Magic	
	(Magic to Stone Steps)	
0098 49	U to (at) House (9)	
0099 05	Cage	
009A 02	Rope	
009B 04	File	
;		
009C 81	LOCNUM = 1	<u>Purple Oracle</u>
009D 0A	Directions: W,E	
009E 4A	E to Bird Room (A)	
009F 43	W to Stone Steps (3)	
;		
00A0 82	LOCNUM = 2	<u>Red Room</u>
00A1 52	Directions: E,U,Magic	
	(Magic to Cellar)	
00A2 47	E to Gully (7)	
00A3 52	U to Attic (12)	
00A4 07	Gold	
;		
00A5 83	LOCNUM = 3	<u>Stone Steps</u>
00A6 70	Directions: U,D,Magic	
	(Magic to Cellar)	
00A7 41	U to Purple Oracle (1)	
00A8 48	D to Royal Hall (8)	
;		
00A9 84	LOCNUM = 4	<u>Blue Den</u>
00AA 61	Directions: N,D,Magic	
	(Magic to Cellar)	
00AB 46	N to Hole (6)	
00AC 4F	D to Oyster-Bed (F)	
;		
00AD 85	LOCNUM = 5	<u>Steel Grate</u>
00AE 21	Directions: N,D	
00AF 55	N to 2-Inch Slit (15)	
00B0 54	D to Tunnel (14)	
;		

;		
00B1	86	LOCNUM = 6 <u>Hole</u>
00B2	00	Directions: None!
;		
00B3	87	LOCNUM = 7 <u>Gully</u>
00B4	0C	Directions: S,W
00B5	4E	S to Grotto (E)
00B6	42	W to Red Room (2)
;		
00B7	89	LOCNUM = 9 <u>House</u>
00B8	2E	Directions: E,S,W,D
00B9	57	E to Forest (17)
00BA	4B	S to Stream (B)
00BB	56	W to Glen (16)
00BC	40	D to Cellar (0)
;		
00BD	8A	LOCNUM = A <u>Bird Room</u>
00BE	0C	Directions: S,W
00BF	54	S to Tunnel (14)
00C0	41	W to Purple Oracle (1)
00C1	01	Bird
;		
00C2	8B	LOCNUM = B <u>Stream</u>
00C3	0F	Directions: N,E,S,W
00C4	49	N to House (9)
00C5	57	E to Forest (17)
00C6	55	S to 2-Inch Slit (15)
00C7	56	W to Glen (16)
;		
00C8	8C	LOCNUM = C <u>Tight Shaft</u>
00C9	30	Directions: U,D
00CA	52	U to Attic (12)
00CB	53	D to S. Pit (13)
;		
00CC	8D	LOCNUM = D <u>N. Pit</u>
00CD	2A	Directions: E,W,D
00CE	51	E to E. Pit (11)
00CF	50	W to Chute (10)
00D0	46	D to Hole (6)
;		
00D1	8E	LOCNUM = E <u>Grotto</u>
00D2	0D	Directions: N,S,W
00D3	43	N to Stone Steps (3)
00D4	4D	S to N. Pit (D)
00D5	47	W to Gully (7)
;		
00D6	97	LOCNUM = 17 <u>Forest</u>
00D7	09	Directions: N,W
00D8	49	N to House (9)
00D9	4B	W to Stream (B)
;		

```

OODA 90          LOCNUM = 10          Chute
OODB 20          Directions: D
OODC 4C          D to Tight Shaft (C)
;
OODD 91          LOCNUM = 11          E. Pit
OODE 35          Directions: N,S,U,D
OODF 4D          N to N. Pit (D)
OOEO 53          S to S. Pit (13)
OOE1 4C          U to Tight Shaft (C)
OOE2 46          D to Hole (6)
;
OOE3 92          LOCNUM = 12          Attic
OOE4 20          Directions: D
OOE5 48          D to Royal Hall (8)
;
OOE6 93          LOCNUM = 13          S. Pit
OOE7 33          Directions: N,E,U,D
OOE8 4D          N to N. Pit (D)
OOE9 51          E to E. Pit (11)
OOEA 52          U to Attic (12)
OOEB 44          D to Blue Den (4)
;
; EGO File ("File of the self")
;   Behaves like any other location, except that the
;   "Directions" word is used for the Most Significant
;   Half of the double precision MOVES counter. This
;   file is initially empty; objects picked up by the
;   adventurer are placed here until they are dropped.
;
OOEC 9F          LOCNUM = 1F          EGO File
OOED 00          M.S.H. of MOVES
OOEE 9F EOCM     End Of Cave Map Flag (a constant)
;
; KIM monitor locations used by KIM-VENTURE
;
OOEF          YSAV    RES 1    Used by LIGHT S/R to save Y-Reg.
;           ; This location is destroyed each
;           ; time ADDOBJ is called -- EOCM
;           ; gets written here.
OOFO          WINDO   RES 7    Display window for LIGHT S/R. Really
;           ; only need six, but for the fact
;           ; that FILMSG keeps unpacking msg's
;           ; till it ends on a whole byte --
;           ; thus clobbering 1 or 2 extras...
OOF7          DIR     RES 1    Direction moved. 0=N,...,5=D.
;
OOF8          TEMP    RES 1    Used by LIGHT and monitor together.
OOFD          LCTR     RES 1    Letter-counter for FILMSG.
OOF9          DISNXM  RES 1    Display-next-message flag. If nonzero,
;           ; FILMSG will add DISNXM to ADL of
;           ; message (POINTR) and start over.

```

	:	START	segment. Begin here using keys	
	:		<u>AD</u> , 0100, <u>GO</u> .	
	:			
0100	D8	START	CLD	PROGRAM START POINT.
0101	A5 45		LDA LOCNUM	Start at preloaded loc.
	:			
	:		NEWLOC segment. Program comes here any	
	:		time a location is entered.	
	:			
0103	85 45	NEWLOC	STA LOCNUM	New location entry.
0105	A2 7B		LDX #SOCM-2	Start-of-cave-map is
				used as starting
				point for file search.
0107	E8		CKLNUM INX	
0108	E8		CKLNLP INX	
0109	B5 00		LDA 0,X	Is this a start-of file?
010B	10 FB		BPL CKLNLP	No. Keep looking.
010D	86 42		STX LOCAD	Yes, save file address,
010F	29 1F		AND #\$1F	and see if it's the
0111	C5 45		CMP LOCNUM	one he moved to ...
0113	D0 F2		BNE CKLNUM	No. Look for next file.
0115	09 A0		ORA #\$A0	Yes. Indicate "visited
0117	95 00		STA 0,X	here" for scoring.
0119	B4 01		LDY 1,X	Get "Directions" word.
011B	29 01		AND #1	Set X-reg for "IN" if
011D	AA		TAX	LOCNUM even; else "AT".
011E	84 3B		STY SGNPST	Signpost = "Directions".
0120	B4 5F		LDY UINMAD,X	Show "You are in"
0122	20 B3 02		JSR FILMSG	or "You are at".
0125	A6 45		LDX LOCNUM	
0127	B4 65		LDY LNAMAD,X	
0129	20 B3 02		JSR FILMSG	Show location name.
012C	A6 45		LDX LOCNUM	
012E	CA		DEX	At Purple Oracle?
012F	D0 12		BNE MVTOBH	No, move to obj-handler.
0131	AD 06 17		LDA TIMER	Yes, so pick
0134	29 0F		AND #\$F	a random magic button.
0136	AA		TAX	
0137	85 46		STA MBUT	Save the button
0139	BD E7 1F		LDA DISCOD,X	and the display-code
013C	85 3C		STA MBCODE	(from monitor) for it.
013E	A0 8F		LDY #ASSMAD	Show "A Sign Says
0140	20 B3 02		JSR FILMSG	Magic Button Is *
0143	A9 0B	MVTOBH	LDA #\$B	Set up for "I See" in
0145	4C 00 02	OBHLNK	JMP OBHNDL	Object-Handler.

```

; MNMVLP (Main Move Loop).  Program comes here
; after each move and stays here till next.
;
0148 A6 41      MNMVLP LDX EGOLAD      Enter here after each move.
014A E6 4C      INC MOVES             Move count.  Overflow?
014C D0 02      BNE MNLOOP            No.
014E F6 01      INC 1,X               Bump MSH of MOVES.
0150 A0 FC      MNLOOP LDY #CUEMAD     Loop here till he moves.
0152 20 B3 02   JSR FILMSG            Show "?", Signpost.
0155 C9 06      CMP #6                Key = 0 - 5? (Dir?)
0157 B0 03      BCS MNLCON
0159 4C A5 17   JMP SPROC             Yes. Do Special Proc.
015C C9 0B      MNLCON CMP #$B         Key = 6 - A? (No-op?)
015E 90 E8      BCC MNMVLP            Yes. Count as a move.
0160 F0 9E      STLINK BEQ START       Key = B. To Browse, act
;                                     ; as if just moved here.
0162 C9 0F      CMP #$F               Key = C,D, or E?
0164 90 DF      BCC OBHLNK            Yes. Handle objects. (Go
;                                     ; via NEWLOC.)
0166 D0 E8      BNE MNLOOP            If key is none of the
;                                     ; above, and not F,
;                                     ; do nothing.
; F key has been hit. Magic Processing.
0168 A9 53      LDA #53               Insert "?" to ask what
016A B5 3C      STA MBCODE             Magic Button is.
016C A0 9E      LDY #MBIMAD
016E 20 B3 02   JSR FILMSG            Ask the question.
0171 A0 E1      LDY #CUEMAD
0173 C5 46      CMP MBUT              Did he hit the right one?
0175 D0 D1      BNE MNMVLP            No. Count as a move.
0177 A9 03      LDA #3                Yes, so magic might work.
0179 A6 45      LDX LOCNUM            In the Cellar?
017B F0 2A      BEQ NEWLNK            Yes. To Stone Steps now.
017D A9 00      LDA #0                Is location number
017F E0 05      CPX #5                higher than 4?
0181 B0 3B      BCS NOJMSG            Yes. Spell won't work.
0183 CA         DEX                  At Purple Oracle?
0184 F0 DA      BEQ STLINK            Yes. Spell not only won't
;                                     ; work, it changes!
0186 EE 3D 00   INC NMBUTS            OK. At Stone Steps, Red
;                                     ; Room, or Blue Dex..
;                                     ; Bump M.B. count, and
0189 D0 1C      BNE NEWLNK            ; go to Cellar (via
;                                     ; MOVER).
;
; MOVER  Processes direction commands (if you
; made it through SPROC).
;
018B A6 42      MOVER  LDX LOCAD       Address current file,
018D B5 01      LDA 1,X               pick up "Directions",
018F A0 FF      LDY #$FF              and init check count.

```

0191 C8	CKNDIR	INY	
0192 4A		LSR	This direction OK?
0193 90 05		BCC CKDLP	No. See if done.
0195 E8		INX	Yes. Bump pointer, and
0196 C4 F7		CPY DIR	; see if this is the
			; desired direction.
0198 F0 09		BEQ DIROK	It is. Go do it.
019A C0 05	CKDLP	CPY #5	Isn't. Tried all dir's?
019C D0 F3		BNE CKNDIR	No, keep on...
019E A0 F7		LDY #CNTMAD	Show "Cannot" and
01A0 4C 22 02	MVMSML	JMP MSGAML	return to main loop.
01A3 B5 01	DIROK	LDA 1,X	Pick up new location
01A5 29 1F		AND #\$1F	; number, get LS 5
			; bits for LOCNUM, and
01A7 4C 03 01	NEWLNK	JMP NEWLOC	Go to new location.
			; OBUSE (Object Use, or Employment)
01AA A4 49	OBUSE	LDY OBJ	
01AC A5 45		LDA LOCNUM	
01AE 88		DEY	Is object Bird?
01AF F0 11		BEQ OBUBRD	Yes. Go use it.
01B1 88		DEY	Is object Rope?
01B2 F0 21		BEQ OBUROP	Yes. Go use it.
01B4 A2 07		LDX #GULLY	
01B6 88		DEY	Is object Rod?
01B7 F0 24		BEQ OFLROD	Yes. Go to File/Rod use.
01B9 A2 05		LDX #STGRAT	
01BB 88		DEY	Is object File?
01BC F0 1F		BEQ OFLROD	Yes. Go to File/Rod use.
01BE A0 85	NOJMSG	LDY #NOJMAD	Show
01C0 D0 DE		BNE MVMSML	"No Joy" (via MOVER).
01C2 C9 06	OBUBRD	CMP #RYHALL	Used Bird at Royal Hall?
01C4 D0 F8		BNE NOJMSG	No -- nothing happens.
01C6 A5 40		LDA DRAGON	Yes. Dragon hungry?
01C8 F0 F4		BEQ NOJMSG	No, dead. No effect.
01CA C8		INY	Yes! "Using" Bird is
01CB 84 40		STY DRAGON	like "feeding him to
01CD A0 BD		LDY #ADDGMS	Dragon!" Show
01CF 20 B3 02		JSR FILMSG	"Dragon Eats Bird".
01D2 4C 7F 02		JMP OBDELE	Go delete Bird.
01D5 C9 06	OBUROP	CMP #HOLE	Used Rope in Hole?
01D7 D0 E5		BNE NOJMSG	No. No effect.
01D9 A9 0D		LDA #NPIT	Yes, so got out to
01DB 10 CA		BPL NEWLNK	N. Pit (via MOVER).

01DD E4 45	OFLROD CPX LOCNUM	Used File at Grate or ; Rod at Gully?
01DF D0 DD	BNE NOJMSG	No. No effect.
01E1 B5 4F	LDA ADOFGR-5,X	Yes. Is Grate open or
01E3 D5 65	CMP LNAMAD,X	is Bridge made?
01E5 F0 D7	BEQ NOJMSG	Yes. No effect.
01E7 95 65	STA LNAMAD,X	No. Open Grate or ; make the Bridge.
01E9 8A	TXA	Show the new state
01EA 10 BB	BPL NEWLNK	of this location.
;		
; DELOBJ (Delete Object) Subroutine.		
; Call with DLOBAD = page zero address		
; of the object to be deleted from file.		
;		
01EC A6 3F	DELOBJ LDX DLOBAD	Point to obj to delete.
01EE B5 01	DOBLP LDA 1,X	Move all files down
01F0 95 00	STA 0,X	one location until
01F2 E8	INX	obj is overwritten.
01F3 E0 EF	CPX #EOCM+1	Done yet?
01F5 D0 F7	BNE DOBLP	No, continue.
01F7 60	RTS	Yes, return.
;		
; OBHNDL (Object-Handler) segment. Entered		
; with A-reg filled with either of key-		
; depressions B,C,D, or E. (Arrival at		
; a location looks like a B-keyin.)		
; B=Browse. Produces list of objects,		
; with no action allowed.		
; C,D,E = Carry, Drop, Employ. Each		
; produces object list, but during		
; list, any key causes action on		
; object currently displayed.		
;		
0200 38	OBHNDL SEC	Change B,C,D, or E to
0201 E9 0D	SBC #\$D	-2,-1,0, or 1.
0203 AA	TAX	
0204 4A	LSR	Set up Y-reg for LOBSCH:
0205 29 01	AND #1	Y=1 (current loc)-B,C.
0207 A8	TAY	Y=0 (EGO file)-D,E.
0208 49 01	EOR #1	Flip state to get "loc-
020A 85 43	STA LINTAX	of-interest-adr-index".
020C E8	INX	Change B,C,D, or E to
020D 86 55	STX SCDU	-1,0,1, or 2 for SCDU.
020F 20 80 17	JSR LOBSCH	Get LOBJAD, no. of obj's.
0212 84 48	STY NOBS	Save no. of obj's for loop.
0214 F0 0F	BEQ MLLINK	If nothing here, done!
;		
; Begin object-handling processing...		

0216	A6 55		LDX SCDU	"Carry" command?
0218	D0 0E		BNE OBHMDS	No. Continue.
021A	A5 47		LDA NOBCRY	Yes, but is he already
021C	C9 04		CMP #4	carrying four things?
022E	D0 08		BNE OBHMDS	No. Continue.
022D	A0 FA	HOWMSG	LDY #HOWMAD	Show "How ? ".
0222	20 B3 02	MSGAML	JSR FILMSG	Display the message.
0225	4C 48 01	MLLINK	JMP MNMVLP	Return to Main Move Loop.
;				
0228	B4 62	OBHMDS	LDY CYMSAD,X	Show "I See-", "Carry-",
022A	20 B3 02		JSR FILMSG	"Drop-", or "Use - ".
;				
022D	C6 48	OBNEXD	DEC NOBS	Showed all obj's yet?
022F	30 F4		BMI MLLINK	Yes. Nothing else to do.
0231	A4 44		LDY LOBJAD	Save addr of this object
0233	84 3F		STY DLOBAD	; in case it's to be
; deleted from the file.				
0235	B6 00		LDX 0,Y	Save the
0237	86 49		STX OBJ	object number.
0239	B4 57		LDY OBMSAD,X	Show the
023B	20 B3 02		JSR FILMSG	object's name.
023E	A4 55		LDY SCDU	Just locking?
0240	30 04		BMI ORN	Yes, display next one.
0242	C9 15		CMP #\$15	Carry/Drop/Use this obj?
0244	D0 04		BNE OBHXQT	Yes. Execute obj-handle.
0246	C6 44	OBN	DEC LOBJAD	Point to next object,
0248	D0 E3		BNE OBNEXD	and show it.
;				
; Execution of object-handling begins:				
024A	88	OBHXQT	DEY	
024B	30 05		BMI OBCARY	Go Carry object.
024D	F0 29		BEQ OBDROP	Go Drop object.
024F	4C AA 01		JMP OBUSE	Go Use object.
;				
; OBCARY (Object-Carrying) segment.				
;				
0252	A0 F7	OBCARY	LDY #CNTMAD	
0254	A6 49		LDX OBJ	
0256	CA		DEX	Is object Bird?
0257	D0 08		BNE OBCDCK	No, see if Dragon.
0259	A5 3E		LDA BURDEN	Yes. Is he carrying
025B	29 28		AND #\$28	the cage and
025D	C9 20		CMP #\$2C	not the rod?
025F	D0 BF		BNE HOWMSG	No. "How carry Bird?"
;				
0261	8A	OBCDCK	TXA	Is obj Dragon (X=\$FF)?
0262	30 BE		BMI MSGAML	Yes. Show "Cannot".
;				
; Finally ready to carry the				
; indicated object...				

Address	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418	Op419
---------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

			; FILMSG (Fill WINDO, display message) Sub-
			; routine. Unpacks and displays a word
			; or series of words, starting at ADL
			; specified by Y-reg at time of call.
			; Message is in page specified by con-
			; tents of POINTR+1. Calls LIGHT S/R.
02AE 18 8A	FLM1	CLC TXA	To display next word,
02B0 65 4A		ADC POINTR	add DISNXM to POINTR,
02B2 A8		TAY	place in Y-reg, and
			call S/R again...
02B3 84 4A	FILMSG	STY POINTR	S/R ENTRY POINT *****
			; Save msg ADL.
02B5 A2 00		LDX #0	Clear letter-counter.
02B7 A0 00		LDY #0	Clear byte pointer.
02B9 86 FE		STX DISNXM	Clear "continue" flag.
02BB 86 FD	MFLOOP	STX LCTR	Save letter-counter.
02BD 18		CLC	C=0 to address FUTBL.
02BE B1 4A	MFLAP	LDA (POINTR),Y	Get next byte, and
02C0 48		PHA	save a copy.
02C1 6A		ROR	Shift in CARRY bit,
02C2 4A		LSR	then move CARRY+MSH
02C3 4A		LSR	to lower part
02C4 4A		LSR	of the byte.
02C5 F0 32		BEQ MSHRPT	MSH=0 means LSH is a
			; repeat pointer.
02C7 C9 01		CMP #1	MSH=1 means LSH is an
02C9 F0 34		BEQ IUBYT	index to IUTBL.
02CB AA		TAX	MSH ≥ 2, so use C + MSH
02CC B5 1F		LDA FUTBL-2,X	to point to char-code.
02CE A6 FD	STMSH	LDX LCTR	Use letter-counter to
02D0 95 F0		STA WINDO,X	put code in window.
02D2 68		PLA	Get copy of current byte.
02D3 E8		INX	Increment and
02D4 86 FD		STX LCTR	save the letter count.
			; (At this point, FILMSG
			; <u>could</u> be done, and a
			; check should be made
			; for "Done 6?". To
			; save 4 bytes, I let it
			; run till ending on a
			; byte boundary....RCL)
02D6 29 OF		AND #\$F	Extract LSH of the byte.
02D8 C9 01		CMP #1	If =1, next byte is IU
02DA F0 26		BEQ IUNXWD	letter code.
02DC AA	STLSH	TAX	Use this byte's LSH as
02DD B5 1F		LDA FUTBL-2,X	FUTBL pointer.
02DF A6 FD		LDX LCTR	Use letter-counter to
02E1 95 F0		STA WINDO,X	put code in window.
02E3 C8 E8		INY INX	Bump both pointers.
02E5 E0 06		CPX #6	Done yet?
02E7 90 D2		BCC MFLOOP	No. Continue.

02E9	A0	C0	DONFIL	LDY	#\$C0	**(\$02EA)=Display speed**
02EB	20	00	00	SHMSG	JSR LIGHT	Make several calls to
02EE	20	00	00		JSR LIGHT	the display/keyboard
02F1	88				DEY	subroutines.
02F2	D0	F7			BNE SHMSG	
02F4	A6	FE			LDX DISNXM	Display another word?
02F6	D0	B6			BNE FLM1	Yes. Go do it.
02F8	60				RTS	No. Return with key (if
					;	any) in A-reg. If no
					;	key hit, A = \$15.
02F9	68		MSHRPT	PLA		The current byte is an
02FA	85	FE		STA	DISNXM	offset to next msg.
02FC	C8			INY		Save it, point to
02FD	10	BC		BPL	MFLOOP	next byte, continue.
					;	
02FF	68		IUBYT	PLA		The current byte is an
0300	10	DA		BPL	STLSH	IUTBL pointer. Use it.
					;	
0302	C8		IUNXWD	INY		Point to next byte.
0303	38			SEC		C=1 will add 16 to FUTBL
0304	B0	B8		BCS	MFLAP	pointer; thus we have
					;	an IUTBL pointer.
					;	
					;	Messages. Starting at a point in page 3
					;	specified by POINTR, the FILMSG S/R
					;	examines this data a half-byte at a
					;	time to extract a 6-character message.
					;	Each half-byte may be one of the
					;	following:
					;	0, meaning "Save the <u>next</u> half-byte. When
					;	the current display is done, use that
					;	value to advance POINTR, and go through
					;	FILMSG again for a new display."
					;	1, meaning "Use the value of the <u>next</u>
					;	half-byte as a pointer into IUTBL."
					;	2 - F, meaning "Use <u>this</u> value as a pointer
					;	into FUTBL."
					;	(See IUTBL and FUTBL at \$002F, \$0021.)
					;	(Key to character-codes is at end of this table.)
0306	35	88	2B		CELLAR	CE LL AR
0309	05	16	EB	16	85 AB 23 85	PURPLO #5 Px UR Px LE OR AC LE
0311	03	FB	54	FF	BA A1 5F	REDRM #3 R ED RC CM x*
0318	07	CD	A9	5F		STSTPS #7 ST ON E
031C	16	52	B8			PEARLS Px EA RL
031F	CD	51	6C			ST EP xS
0322	F3	21	35		CAGE	C AG xE
0325	F1	27	85		FILE	F xI LE
0328	F3	6E	D5		CHUTE	C HU TE
032B	05	11	B7	41 35	BRAGM	#5 Bx RI DG xE
0330	08	23	BA	CC		#8 AC RO SS
0334	0C	FA	16	59	OPNGRM	#C O Px EN
0338	F1	3E	88	18	GULLY	G xU LL Yx

033C 04 CD 55 8F
 0340 13 B2 D5
 0343 FB A1 65
 0346 F1 3A 84
 0349 F3 21 35
 034C F4 A9 5F
 034F 2F 6A 85
 0352 07 FD 71 36
 0356 DE 99 58
 0359 FC 62 12 DF
 035D 05 1B 10 79 36
 0362 04 FC 87 DF
 0366 09 79 FD 65
 036A OD A1 8C D5 BF
 036F 08
 0370 CD B5 21
 0373 51 AF 16 7D
 0377 F1 01 15 4F
 037B 04 BA 18 28 F6 28 8F
 0382 2F 13 85
 0385 9A F1 4A 18
 0389 EF BF 2D
 038C EF BF 79
 038F 08 2F C7 13
 0393 91 AF 16 7D
 0397 07 FC 21 8C
 039B F2 DD 73
 039E 05 15 21 37 3F
 03A3 05 11 ED DA 9F
 03A8 F7 CF 1D
 03AB F6 AE C5
 03AE FE C5 F1
 03B1 07 62 8D 54
 03B5 7F C5 51
 03B8 06 11 18 FD 65
 03BD OE
 03BE 4B 21 3A 9F
 03C2 04 C3 2B 54
 03C6 09 AE DF 11 18
 03CB 04 F5 2D CF
 03CF OD 87 DD 85
 03D3 OF F1 18 E5
 03D7 F4 BA 16 10
 03DB 04
 03DC F1 17 B4
 03DF FB AA 15
 03E2 F4 59
 03E4 FF BA 4F
 03E7 32 BB 18 10
 03EB C1 AF 16 7D
 03EF 13 BA DD AF

STGRAT #4 ST EE L
 Gx RA TE
 ROPE R OP xE
 GOLD G xO LD
 CAGE C AG xE
 DONMAD D ON E
 HOLE A HO LE
 TSHAFT #7 T IG xH
 TUNNEL TU NN EL
 S HA Fx T*
 SLIT #5 2x -x IN CH
 #4 S LI T
 #9 IN T HE
 OYSTRB #D OY xS TE R*
 #8
 STREAM ST RE AM
 EPIT E. x Px IT
 - xB xE D
 RYHALL #4 RO Yx AL H AL L
 GLEN A Gx LE
 NOJOY NO J xO Yx
 URAT U R AT
 URIN U R IN
 ASSMAD #8 A SI Gx
 NPIT N. x Px IT
 #7 S AY xS
 ATTIC A TT IC
 MBISAD #5 Mx AG xI C
 #5 Bx UT TO N*
 I S @x
 HOUSE H OU SE
 USE U SE -
 HBDRCN #7 HA LT ED
 ISEE I SE E-
 #6 Bx Yx T HE
 ADDCMS #E (Change to #5 for
 ; Scared Out...)
 ADDRAG DR AG xO N*
 #4 SC AR ED
 #9 OU T Bx Yx
 #4 E AT S
 #D LI TT LE
 BLUDEN #F B xL UE
 DROP D RO Px -x
 BIRDRM #4
 BIRD B xI RD
 R OO Mx
 D EN
 ROD RO D
 CARRY CA RR Yx -x
 SPIT S. x Px IT
 GROTTTO Gx RO TT O*

03F3 12 AB 5C DF
 03F7 32 99 AD
 03FA 6A 17
 03FC F1 9F FF 1C

FOREST Fx OR ES T*
 CNTMAD CA NN OT
 HOWMAD HO Wx
 CUEMAD ? x &x

; Key to characters used in right-hand column
 of above table;

; Letter or space -- the FUTBL 4-bit code
 for that letter or space.

; Letter, dash, "?", ".", or 2; followed
 by "x" -- the IUTBL 8-bit code for that
 character.

; "@x" -- the IUTBL 8-bit code for the
 character stored in IUTBL (by the
 NEWLOC and MNMVLP segments) as
 part of the Magic Button message.

; "&x" -- the 8-bit code for the Signpost
 character stored in IUTBL (by the
 NEWLOC segment) as part of CUE msg.

; #n -- the number, n, of bytes (in hex) to
 advance POINTR in order to point to
 the next successive message.

; * -- a "wasted" half-byte

;

; LOBSCH (Last Object Search) subroutine.

; Finds, and saves in LOBJAD, the address
 of the last object in a file; also
 counts, and returns in Y-reg, the number
 of objects in the file. File to search
 is ECO file if called with Y=0; is
 file at LOCAD if called with Y=1.

; LOBSCH LDX EGOLAD,Y Get pointer to file.
 LDY #\$FF Init object-count.

; OBFIND INX
 STX LOBJAD Save addr of last obj.
 LDA 1,X Set up to test bits
 ASL 7 & 6 of each location.
 BMI OBFIND b6=1. Not an object.
 INY Bump object-count.
 BCC OBFIND b7=0. An object. Continue.
 RTS b7=1. End of file. Done.

1780 B6 41
 1782 A0 FF

1784 E8
 1785 86 44
 1787 B5 01
 1789 0A
 178A 30 F8
 178C C8
 178D 90 F5
 178F 60

```

;
; ADDOBJ (Add Object) subroutine. Called to
; add a dropped object to a location file,
; or a picked-up object to EGO file.
; LINTAX is the pointer to the address of
; the location of interest: 0 for EGO,
; 1 for file specified by LOCAD. Calls
; LOBSCH subroutine. Object to be added
; is specified by contents of OBJ.
;
1790 A4 43 ADDOBJ LDY LINTAX      Point to file of interest.
1792 20 80 17 JSR LOBSCH      Find last obj's address.
1795 A2 EE     LDX #EOCM      Start at End of Cave Map.
1797 B5 00 AOBLP LDA 0,X       Move all files up one
1799 95 01     STA 1,X        location to make room
179B CA       DEX            for the object.
179C E4 44     CPX LOBJAD     Done yet?
179E D0 F7     BNE AOBLP      No. Keep moving.
17A0 A5 49     LDA OBJ        Yes, store object just
17A2 95 01     STA 1,X        above last object in
17A4 60        RTS           the file; return.
;
; SPROC (Special Processing) segment.
; Entered from Main Move Loop (MNMVLP)
; following a "direction" command, this
; code takes care of any special pro-
; hibitions against moving in the com-
; manded direction. (Examples -- can't
; go through a steel grate, or past a
; dragon.) Possible exits from SPROC
; are: to MOVER, if no problems with
; the commanded direction,
; to HOWMSG, if "How ? " is to be
; shown to indicate improper
; conditions for the move, or
; to MSGAML, showing "Halted By
; The Dragon", if appropriate.
;
17A5 A8 SPROC TAY
17A6 84  F7   STY DIR          Save direction for MOVER.
17A8 A6 45   LDX LOCNUM
17AA B5 65   LDA LNAMAD,X      If at grate (or gully),
17AC D5 4F   CMP ADOPGR-5,X    is grate open (or is
; bridge made)?
17AE F0 08   BEQ SPATS        Yes, move is OK.
17B0 E0 05   CPX #(ADGRM-LNAMAD) No. At closed
; grate?
17B2 F0 24   BEQ SPCHKD       Yes, disallow Down.
17B4 E0 07   CPX #(ADGYM-LNAMAD) At bridgeless
; gully?
17B6 F0 22   BEQ SPCHKW       Yes, disallow West.

```

17B8 E0 0C	SPATS	CPX #(ADTSM-LNAMAD)	At shaft?
17BA D0 04		BNE SPATSS	
17BC A5 47		LDA NOBCRY	Yes, carrying anything?
17BE D0 18		BNE SPCHKD	Yes, disallow Down.
		:	
17C0 E0 03	SPATSS	CPX #(ADSSM-LNAMAD)	At steps?
17C2 D0 04		BNE SPATRH	
17C4 A5 3E		LDA BURDEN	Yes, carrying Gold?
17C6 30 11		BMI SPCHKU	Yes, disallow Up.
		:	
17C8 E0 08	SPATRH	CPX #(ADRHM-LNAMAD)	At Royal Hall?
17CA D0 15		BNE SPCONT	
17CC A5 40		LDA DRAGON	Yes, is Dragon there?
17CE F0 11		BEQ SPCONT	
17D0 88		DEY	Yes, but going East
17D1 F0 0E		BEQ SPCONT	is OK. Continue.
17D3 A0 B1		LDY #HBDMS	All other directions,
17D5 4C 22 02		JMP MSGAML	"Halted by Dragon."
		:	
17D8 88	SPCHKD	DEY	Check for Down,
17D9 88	SPCHKU	DEY	for Up, or
17DA C0 03	SPCHKW	CPY #3	for West.
17DC D0 03		BNE SPCONT	Other directions are OK.
17DE 4C 20 02		JMP HOWMSG	Disallowed direction
		:	produces "How ? ".
		:	
17E1 4C 8B 01	SPCONT	JMP MOVER	Continue Move process.
		:	
		:	
		:	
17E4 00	:	These three bytes	
17E5 00	:	are left spare for	
17E6 00	:	user expansion....	

ARESCO
BOX 1142
COLUMBIA
MD 21044
(301) 730-5186

December, 1979

Robert C. Leedom
14069 Stevens Valley Ct
Glenwood, MD 21738


```

; K - V SCORE  © Copyright R.C.Leedom 1979
; The K-V SCORE program is to be loaded
; immediately following a KIM-VENTURE
; game, and run starting at location
; $100. K-V SCORE will provide a rating
; (which may be from Class A all the
; way down to J, or -- at the bottom --
; Class O), and a count of the moves
; made by the player (up to 9999).
0100 4C 24 01 KVSCOR JMP BGNSCR

;
; LOCSCH (Location Search) Subroutine.
; Created from KIM-VENTURE's NEWLOC,
; this S/R (when called with A-reg =
; location number) will search for the
; location file and will return LOCAD
; in the X-register.
0103 85 45 LOCSCH STA LOCNUM      Save location number.
0105 A2 7B      LDX #SOCM-2      Start-of-cave-map is
;                                ; used as starting
;                                ; point for file search.

0107 E8          CKLNUM INX
0108 E8          CKLNLP INX
0109 B5 00      LDA 0,X          Is this a start-of-file?
010B 10 FB      BPL CKLNLP      No. Keep looking.
010D 86 42      STX LOCAD       Yes, save file address,
010F 29 1F      AND #$1F        and see if it's the
0111 C5 45      CMP LOCNUM      one we want ...
0113 D0 F2      BNE CKLNUM      No. Look for next file.
0115 60          RTS           Yes. Done, so return.

;
; VISCHK (Visit Check) Subroutine. Call with
; A-reg = location number. S/R will re-
; turn A-reg <0 if location was visited,
; else A-reg >0.
0116 20 03 01 VISCHK JSR LOCSCH Go get LOCAD in X-reg.
0119 B5 00      LDA 0,X          Now get header word of
011B 0A 0A      ASL ASL         location file, shifted
011D 60          RTS           to show "visit" bit,
;                                and return.

;
011E 6D 58 5C SCRMSG DATA 6D 58 5C Data for "SCORE " msg.
0121 50 79 00      DATA 50 79 00

;
; BGNSCR (Begin Scoring) segment. (Main prog.)
;
0124 A2 06      BGNSCR LDX #6    Display the six
0126 BD 1D 01 SCMLP  LDA SCRMSG-1 characters of the
0129 95 EF      STA WINDO-1,X   score message:
012B CA          DEX           save them in the
012C D0 F8      BNE SCMLP      window, indicate
012E 86 FE      STX DISNXM     "no more displays",
0130 20 E9 02 JSR DONFIL      and call a few LIGHTs.

```

```

; MVCONV (Move Conversion) segment.  Converts
; the double precision move counter to
; a decimal number (up to four digits).

```

```

0133 18      MVCONV CLC
0134 F8      SED                      Set decimal mode.
0135 A9 00    LDA #0                  Clear Binary Coded
0137 85 61    STA BCDLSH              Decimal, Least and
0139 85 62    STA BCDMSH              Most Signif. Halves.
013B A5 4C    LDA MOVES               Is LSH = 0?
013D F0 10    BEQ MSADD              Yes, go add up MSH.

```

```

013F A5 61    LSADD LDA BCDLSH        For LSH, a double
0141 69 01    ADC #1                  precision add
0143 85 61    STA BCDLSH              of one count
0145 A5 62    LDA BCDMSH              for each unit
0147 69 00    ADC #0                  of the LSH of
0149 85 62    STA BCDMSH              the move counter.
014B C6 4C    DEC MOVES
014D D0 F0    BNE LSADD

```

```

014F A6 41    MSADD LDX EGOLAD        Get MSH of the move
0151 B5 01    LDA 1,X                 counter, save it,
0153 85 60    STA MOVMSH              and if zero, we
0155 F0 10    BEQ DSPFIL              are done...

```

```

0157 A5 61    MSAD1 LDA BCDLSH        For MSH, a double
0159 69 56    ADC #56                 precision add of
015B 85 61    STA BCDLSH              256 counts
015D A5 62    LDA BCDMSH              for each unit
015F 69 02    ADC #2                  of the MSH of
0161 85 62    STA BCDMSH              the move counter.
0163 D6 01    DEC 1,X
0165 D0 F0    BNE MSAD1

```

```

;
; DSPFIL (Display Fill) segment.  Fills the
; display window with digits corresponding
; to score, and blanks (up to) two leading
; zeroes.

```

```

0167 D8      DSPFIL CLD
0168 A5 61    LDA BCDLSH              For LSH of score,
016A 29 0F    AND #$F                 get lower digit and
016C AA      TAX                      corresponding segment
016D BD E7 1F LDA DIGCOD,X            code from monitor, and
0170 85 F5    STA WINDO+5             put in display window.
0172 A5 61    LDA BCDLSH              Similarly, get upper
0174 4A 4A    LSR LSR                 digit of LSH of
0176 4A 4A    LSR LSR                 score, use to get
0178 AA      TAX                      segment code from
0179 BD E7 1F LDA DIGCOD,X            KIM monitor, and put
017C 85 F4    STA WINDO+4             in display window.

```

017E A5 62	LDA BCDMSH	For MSH of score,
0180 4A 4A	LSR LSR	first get upper digit.
0182 4A 4A	LSR LSR	
0184 A8	TAY	Is it zero?
0185 F0 04	BEQ LZ1BNK	Yes, blank it.
	;	
0187 AA	TAX	Use nonzero upper digit
0188 BD E7 1F	LDA DIGCOD,X	to get segment code.
018B 85 F2	LZ1BNK STA WINDOW+2	Put MS digit in window.
018D A5 62	LDA BCDMSH	Get next most signif.
018F 29 0F	AND #\$F	digit and
0191 AA	TAX	save it.
0192 98	TYA	Was MS digit zero?
0193 D0 03	BNE NOBNK2	No, so don't blank
	;	this digit.
0195 8A	TXA	Yes, is this one zero?
0196 F0 03	BEQ LZBNK2	Yes -- both zero! Go
	;	blank this one too.
0198 BD E7 1F	NOBNK2 LDA DIGCOD,X	Get code for 2nd MSD.
019B 85 F3	LZBNK2 STA WINDOW+3	Fill the remaining
	;	slot of the window.
	;	CLASS segment. The remaining code
	;	determines the player's classification
	;	based on what was accomplished in the
	;	course of the game.
019D A2 71	CLASS LDX #\$71	Class F if
019F A5 45	LDA LOCNUM	
01A1 C9 06	CMP #6	in the hole
01A3 D0 06	BNE CELSCH	
01A5 A5 3E	LDA BURDEN	
01A7 29 04	AND #4	without the rope.
01A9 F0 28	BEQ WSLNK	Go show "F".
	;	
01AB A9 00	CELSCH LDA #0	Cellar search to see
01AD 85 63	STA OBCELR	if any treasures
01AF 20 03 01	JSR LOCSCN	have been left
01B2 AC 01	LDY #1	here...
01B4 20 80 17	JSR LOBSCH	
01B7 84 48	STY NOBS	Anything here?
01B9 F0 15	BEQ CVLINK	No, check cave visits.
01BE B4 00	OBCSET IDY 0,X	Yes, so set up
01BD A5 63	LDA OBCELR	OBCELR which will
01BF 19 40 00	ORA OBJMSK-1,Y	have a bit set for
01C2 85 63	STA OBCELR	each object left
01C4 CA	DEX	in the cellar.
01C5 C6 48	DEC NOBS	
01C7 D0 F2	BNE OBCSET	
01C9 A2 39	LDX #\$39	Class C if only one
01CB 0A	ASL	treasure here.
01CC 30 08	BMI PICLNK	Pearls. See if Gold too.
01CE B0 03	BCS WSLNK	No Pearls, Gold only.
01D0 4C 12 02	CVLINK JMP CAVIS	Neither, check cave visits.
01D3 4C 64 02	WSLNK JMP WINSET	Go set window with class.
01D6 4C 00 02	PICLNK JMP PICLNK	Continue cellar check.

; Note that 1D9 - 1FF not used. Cellar
 ; check continues in Page 2... At
 ; this point we have verified that
 ; the Pearls are there and are
 ; testing for Gold. A-reg has been
 ; preloaded with Class C.

0200 90 D1	PIC	BCC WSLNK	Pearls only. Class C.
0202 A2 7C		LDX #\$7C	Have placed both
0204 A5 62		LDA BCDMSH	treasures in cellar,
0206 D0 CB		BNE WSLNK	but unless done in
0208 A9 40		LDA #\$40	less than 41 moves,
020A C5 61		CMP BCDLSH	this is only
020C 90 C5		BCC WSLNK	Class B.
020E A2 77		LDX #\$77	Class A for both in
0210 D0 C1		BNE WSLNK	cellar, moves ≤ 40 !

0212 A9 02	CAVIS	LDA #2	No treasures returned.
0214 20 16 01		JSR VISCHK	Visited Red Room?
0217 10 1A		BPL DRAGCK	No.
0219 A9 0F		LDA #\$F	
021B 20 16 01		JSR VISCHK	Visited Oyster-Bed?
021E 10 13		BPL DRAGCK	No.

; Have found (but not recovered) both
 ; treasures, so at least Class E.
 ; See if visited all rooms of
 ; caverns to earn Class D....

0220 A0 12		LDY #\$12	
0222 98	VISCLP	TYA	
0223 20 16 01		JSR VISCHK	Visited this one?
0226 30 04		BMI NXVCLP	Yes, keep checking.
0228 A2 79		LDX #\$79	No, missed one, so
022A D0 A7		BNE WSLNK	show Class E.
022C 88	NXVCLP	DEY	Checked 0 thru \$12?
022D 10 F3		BPL VISCLP	Not yet.
022F A2 5E		LDX #\$5E	Yes, and all were visited,
0231 D0 A0		BNE WSLNK	so show Class D.

; In the code below, no qualifications
 ; have yet been met, so we'll
 ; first see if Class G has been
 ; earned either by scaring off the
 ; Dragon or by using F-key....

0233 A2 3D	DRAGCK	LDX #\$3D	
0235 A5 40		LDA DRAGON	
0237 F0 9A		BEQ WSLNK	Dragon is gone!
0239 A5 3D		LDA NMBUTS	
023B D0 96		BNE WSLNK	F-key used correctly!

; Continuing, let's see if he at
 ; least got into the caverns....

023D A9 14	TUNCK	LDA #\$14	
023F 20 16 01		JSR VISCHK	
0242 A2 76		LDX #\$76	
0244 A8		TAY	Visited Tunnel?
0245 30 8C		BMI WSLNK	Yes, show Class H.
		:	
		:	; Well, did he even get into the
		:	cellar of the house....?
0247 A9 90		LDA #0	
0249 20 16 01		JSR VISCHK	
024C A2 06		LDX #6	
024E A8		TAY	Visited Cellar?
024F 30 82		BMI WSLNK	Yes, show Class I.
		:	
		:	; OK, maybe he forgot he could use
		:	up and down as directions.
		:	But did he do all the exploring
		:	possible with just N,E,S, and W?
0251 A0 C4		LDY #4	
0253 B9 20 02	ABOVLP	LDA VISTBL,Y	Visited House, Glen,
0256 20 16 01		JSR VISCHK	Slit, Forest, Grate?
0259 30 04		RMI ABVCON	
025B A2 3F		LDX #\$3F	No, missed one --
025D D0 05		BNE WINSET	show Class O.
		:	
025F 88	ABVCON	DEY	Checked all 5 yet?
0260 10 F1		RPL ABOVLP	No, keep checking.
0262 A2 1E		LDX #\$1E	Yes, show Class J.
		:	
		:	
0264 86 F0	WINSET	STX WINDC	Put class in window,
0266 A9 40		LDA #\$40	put dash after
0268 85 F1		STA WINDO+1	that, and
026A 20 00 00	END	JSR LIGHT	endlessly show
026D 4C 6A 02		JMP END	Class & Moves.
		:	
		:	
0270 05 39 15	VISTBL	DATA 05 09 15	Table of places to
0273 16 17		DATA 16 17	visit above-ground.

KIM-VENTURE MEMORY MAP

0000	LIGHT S/R
0020	
0021	Character table
003C	
003D	Variables
004C	
004D	Constants
007C	
007D	Cave Map
00EB	
00EC	EGO File
00EE	
00EF	KIM Monitor variables -- some used by KIM-VENTURE
00FF	

0100	START segment
0102	
0103	NEWLOC segment
0147	
0148	MNMVLP segment
018A	
018B	MOVER segment
01A9	
01AA	OBUSE segment
01EB	
01EC	DELOBJ S/R
01F7	
01F8	Stack
01FF	

0200	OBHNDL segment
0251	
0252	OBCARY segment
0277	
0278	OBDROP segment
02AD	
02AE	FILMSG S/R (Entry: 02B3)
0305	
0306	Messages
03FF	

1780	LOBSCH S/R
178F	
1790	ADEOBJ S/R
17A4	
17A5	SPROC segment
17E3	
17E4	(spare)
17E6	

DEC 1979 RCL

