

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.

<u>ID</u>	<u>Loads</u>	<u>Time</u>
-----------	--------------	-------------

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>GO</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.

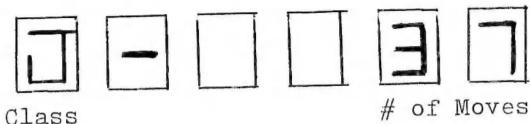
The change can be done by using the key sequence 'AD', 0, 3, B, D, /DA, 0, E. Then begin the game again using the key sequence specified in step 1 ('AD', 0, 1, 0, 0, /GO).

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



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, ANDed 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 ADT, 0100, 1601.

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.

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 extra-ordinaire" to the player with the least number of moves.

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 desireable 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
0004 A9 7F
0006 8D 41 17
0009 A2 09
000B 84 FC    LITELP STY TEMP     Start with leftmost
000D B9 F0 00  LDA WINDO,Y    character.
0010 20 4E 1F  JSR CONVD+6   Get char to be shown.
0013 C8        INY             Use KIM monitor subr.
0014 C0 06        CPY #6       Next char on right ...
0016 90 F3        BCC LITELP   Done all six yet?
                                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
;
0021 77    FUTBL  2      A
0022 39        3      C
0023 5E        4      D
0024 79        5      E
0025 76        6      H
0026 06        7      I
0027 38        8      L
0028 54        9      N
0029 5C        A      O
002A 50        B      R
002B 6D        C      S
002C 78        D      T
002D 1C        E      U
002E 00        F      (blank)
002F 40    IUTBL  10     - (dash)
0030 7C        11     B
0031 71        12     F
0032 3D        13     G

```

; (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	MBUTP	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 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	LNAMEAD	+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 LOCTNUM = 15 2-Inch Slit
008C 05 Directions: N,S
008D 4B N to Stream (B)
008E 45 S to Steel Grate (5)
;
008F 96 LOCTNUM = 16 Glen
0090 02 Directions: E
0091 4B E to Stream (B)
;
0092 8F LOCTNUM = F Oyster-Bed
0093 10 Directions: U
0094 44 U to Blue Den (4)
0095 06 Pearls
;
0096 80 LOCTNUM = 0 Cellar
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 LOCTNUM = 1 Purple Oracle
009D 0A Directions: W,E
009E 4A E to Bird Room (A)
009F 43 W to Stone Steps (3)
;
00A0 82 LOCTNUM = 2 Red Room
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 LOCTNUM = 3 Stone Steps
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 LOCTNUM = 4 Blue Den
00AA 61 Directions: N,D,Magic
 (Magic to Cellar)
00AB 46 N to Hole (6)
00AC 4F D to Oyster-Bed (F)
;
00AD 85 LOCTNUM = 5 Steel Grate
00AE 21 Directions: N,D
00AF 55 N to 2-Inch Slit (15)
00B0 54 D to Tunnel (14)
;

;
 00B1 86 LOCTNUM = 6 Hole
 00B2 00 Directions: None!

;
 00B3 87 LOCTNUM = 7 Gully
 00B4 0C Directions: S,W
 00B5 4E S to Grotto (E)
 00B6 42 W to Red Room (2)

;
 00B7 89 LOCTNUM = 9 House
 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 LOCTNUM = A Bird Room
 00BE CC Directions: S,W
 00BF 54 S to Tunnel (14)
 00CO 41 W to Purple Oracle (1)
 00C1 01 Bird

;
 00C2 8B LOCTNUM = B Stream
 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 LOCTNUM = C Tight Shaft
 00C9 30 Directions: U,D
 00CA 52 U to Attic (12)
 00CB 53 D to S. Pit (13)

;
 00CC 8D LOCTNUM = D N. Pit
 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 LOCTNUM = E Grotto
 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 LOCTNUM = 17 Forest
 00D7 09 Directions: N,W
 00D8 49 N to House (9)
 00D9 4B W to Stream (B)

;

OODA 90 LOCTNUM = 10 Chute
 OODB 20 Directions: D
 OODC 4C D to Tight Shaft (C)
 ;
 OODD 91 LOCTNUM = 11 E. Pit
 OODE 35 Directions: N,S,U,D
 OODF 4D N to N. Pit (D)
 OOE0 53 S to S. Pit (13)
 OOE1 4C U to Tight Shaft (C)
 OOE2 46 D to Hole (6)
 ;
 OOE3 92 LOCTNUM = 12 Attic
 OOE4 20 Directions: D
 OOE5 48 D to Royal Hall (8)
 ;
 OOE6 93 LOCTNUM = 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.
 ;
 OOE_C 9F LOCTNUM = 1F EGO File
 OOE_D 00 M.S.H. of MOVES
 OOE_E 9F EO_{CM} End Of Cave Map Flag (a constant)
 ;
 ; KIM monitor locations used by KIM-VENTURE
 ;
 OOF_F YSAV RES 1 Used by LIGHT S/R to save Y-Reg.
 ; This location is destroyed each
 ; time ADDOBJ is called -- EO_{CM}
 ; gets written here.
 OOF_O 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.
 ;
 OOF_C TEMP RES 1 Used by LIGHT and monitor together.
 OOF_D LCTR RES 1 Letter-counter for FILMSG.
 OOF_E 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
 ; AD, 0100, GO.
 ;
 0100 D8
 0101 A5 45
 ;
 START CLD PROGRAM START POINT.
 LDA LOCNUM Start at preloaded loc.
 ;
 ; NEWLOC segment. Program comes here any
 ; time a location is entered.
 ;
 0103 85 45
 0105 A2 7B
 NEWLOC STA LOCNUM New location entry.
 LDX #SOCM-2 Start-of-cave-map is
 ; used as starting
 ; point for file search.
 ;
 0107 E8 CKLNUM INX Is this a start-of file?
 0108 E8 CKLNLP INX No. Keep looking.
 0109 B5 00 LDA O,X Yes, save file address,
 010B 10 FB BPL CKLNLP and see if it's the
 010D 86 42 STX LOCAD one he moved to ...
 010F 29 1F AND #\$1F No. Look for next file.
 0111 C5 45 CMP LOCNUM Yes, Indicate "visited"
 0113 D0 F2 BNE CKLNUM here" for scoring.
 0115 09 A0 ORA #\$AO Get "Directions" word.
 0117 95 00 STA O,X Set X-reg for "IN" if
 0119 B4 01 LDY 1,X LOCNUM even; else "AT".
 011B 29 01 AND #1 Signpost = "Directions".
 011D AA TAX Show "You are in"
 011E 84 3B STY SGNPST or "You are at".
 0120 B4 5F LDY UINMAD,X
 0122 20 B3 02 JSR FILMSG
 0125 A6 45 LDX LOCNUM
 0127 B4 65 LDY LNAMAD,X
 0129 20 B3 02 JSR FILMSG
 012C A6 45 IDX LOCNUM
 012E CA DEX
 012F D0 12 BNE MVTOBH
 0131 AD 06 17 LDA TIMER
 0134 29 0F AND #\$F
 0136 AA TAX
 0137 85 46 STA MBUT
 0139 BD E7 1F LDA DIGCOD,X
 013C 85 3C STA MBCODE
 013E A0 8F LDY #ASSMAD
 0140 20 B3 02 JSR FILMSG
 0143 A9 0B MVTOBH LDA #\$B
 0145 4C 00 02 OBHLNK JMP OBHNDL

; 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 OF 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 85 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 Did he hit the right one?
 0173 C5 46 CMP MBUT No. Count as a move.
 0175 D0 D1 BNE MNMVLP Yes, so magic might work.
 0177 A9 03 LDA #3 In the Cellar?
 0179 A6 45 LDX LOCTNUM
 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 Den.
 ; Bump M.B. count, and
 ; 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 AO 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 FO 09		BEQ DIROK	It is. Go do it.
019A CO 05	CKDLP	CPY #5	Isn't. Tried all dir's?
019C DO F3		BNE CKNDIR	No, keep on...
019E AO 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
01A7 4C 03 01	NEWLNK	JMP NEWLOC	; bits for LOCNUM, and
		;	Go to new location.
		;	
	OBUSE	(Object Use, or Employment)	
		;	
01AA A4 49	OBUSE	LDY OBJ	Is object Bird?
01AC A5 45		LDA LOCNUM	Yes. Go use it.
01AE 88		DEY	Is object Rope?
01AF FO 11		BEQ OBUBRD	Yes. Go use it.
01B1 88		DEY	
01B2 FO 21		BEQ OBUROP	Is object Rod?
01B4 A2 07		LDX #GULLY	Yes. Go to File/Rod use.
01B6 88		DEY	
01B7 FO 24		BEQ OFLROD	Is object File?
01B9 A2 05		LDX #STGRAT	Yes. Go to File/Rod use.
01BB 88		DEY	Show
01BC FO 1F		BEQ OFLROD	"No Joy" (via MOVER).
01BE AO 85	NOJMSG	LDY #NOJMAD	
01C0 DO DE		BNE MVMSML	
		;	
01C2 C9 26	OBUBRD	CMP #RYHALL	Used Bird at Royal Hall?
01C4 DO F8		BNE NOJMSG	No -- nothing happens.
01C6 A5 40		LDA DRAGON	Yes. Dragon hungry?
01C8 FO F4		BEQ NOJMSG	No, dead. No effect.
01CA C8		INY	Yes! "Using" Bird is
01CB 84 40		STY DRAGON	like "feeding him to
01CD AO 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 DO 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 DO DD	BNF NOJMSG	No. No effect.
01E1 B5 4F	LDA ADOPGR-5,X	Yes. Is Grate open or is Bridge made?
01E3 D5 65	CMP LNAMAD,X	
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	DOBLLP 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 DOBLLP	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 OF	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	;	;
022A 20 B3 02		LDY CYMSAD,X	Show "I See-", "Carry-",
		JSR FILMS:	"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 3F		LDA BURDEN	Yes. Is he carrying
0258 29 28		AND #\$28	the cage and
025D C9 20		CMP #\$20	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...	;

0264 E6 47	OBOKCY	INC NOBCRY	OK to carry object.
0266 A5 3E		LDA BURDEN	Bump carry count,
0268 15 4D		ORA OBJMSK,X	and indicate
026A 85 3E		STA BURDEN	what's being carried.
026C 20 90 17		JSR ADDOBJ	Add obj to EGO file.
026F C6 41	OBDELL	DEC EGOLAD	Move everything down 1,
0271 20 EC 01		JSR DELOBJ	and delete object
		;	from location file.
0274 A0 4C	DONMSG	LDY #DONE	
0276 DO AA		BNE MSGAML	Show "Done" message
		;	and return to
		;	Main Move Loop.
		;	
		;	
		OBDROP (Object-Dropping) segment.	
0278 E6 41	OBDROP	INC EGOLAD	Move everything up 1,
027A 20 90 17		JSR ADDOBJ	and add object to
027D E6 3F		INC DLQBAD	location file.
027F C6 47	OBDELE	DEC NOBCRY	Delete object from
0281 A6 49		LDX OBJ	EGO file,
0283 A5 3E		LDA BURDEN	indicate one less
0285 38		SEC	object carried,
0286 F5 4C		SBC OBJMSK-1,X	and remove
0288 85 3E		STA BURDEN	"object-flag" from
028A 20 EC 01		JSR DELOBJ	Burden list.
		;	
028D A5 40		LDA DRAGON	Is Dragon alive&hungry?
028F 1C E3		BPL DONMSG	No. All done.
		;	
0291 A6 49		LDX OBJ	Was Bird
0293 CA		DEX	just dropped?
0294 DO DE		BNE DONMSG	
0296 A5 45		LDA LOCTNUM	Yes, are we
0298 C9 08		CMP #8	at Royal Hall?
029A DO D8		BNE DONMSG	
029C A9 85		LDA #DRAGAD	Yes, so Dragon is
029E 85 3F		STA DLQBAD	scared off.
02A0 86 40		STX DRAGON	
02A2 A9 05		LDA #5	Change msg length
02A4 8D BD 03		STA ADDGMS	so proper msg
02A7 A0 BD		LDY #ADDGMS	is shown, and
02A9 20 B3 02		JSR FILMSG	go show it.
02AC F0 C1		BEQ OBDELL	Delete Dragon.

; 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 PTR+1. Calls LIGHT S/R.
 02AE 18 8A FLM1 CLC TXA To display next word,
 02B0 65 4A ADC POINTR add DISNMX 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 DISNMX 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 IUBYD index to IUTBL.
 02CB AA TAX MSH \geq 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
 ; could 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)
 ; Extract LSH of the byte.
 ; If =1, next byte is IU
 ; letter code.
 ;
 02D6 29 0F AND #\$F
 02D8 C9 01 CMP #1
 02DA F0 26 BEQ IUNXWD
 ;
 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 E8INY 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	SHMSG	JSR LIGHT	Make several calls to
02EE	20	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	PURPLO	#5 Px UR Px LE OR AC LE
0311	03	FB	54	REDRM	#3 R ED RC OM x*
0318	07	CD	A9	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	BRAGM	#5 Bx RI DG xE
0330	08	23	BA		#8 AC RO SS
0334	0C	FA	16	OPNGRM	#C O Px EN
0338	F1	3E	88	GULLY	G xU LL Yx

033C	04	CD	55	8F	STGRAT	#4	ST	EE	L
0340	13	B2	D5		Gx	RA	TE		
0343	FB	A1	65		ROPE	R	OP	xE	
0346	F1	3A	84		GOLD	G	x0	LD	
0349	F3	21	35		CAGE	C	AG	xE	
034C	F4	A9	5F		DONMAD	D	ON	E	
034F	2F	6A	85		HOLE	A	HO	LE	
0352	07	FD	71	36	TSHAFT	#7	T	IG	xH
0356	DE	99	58		TUNNEL	TU	NN	EL	
0359	FC	62	12	DF		S	HA	Fx	T*
035D	05	1B	10	79	SLIT	#5	2x	-x	IN CH
0362	04	FC	87	DF		#4	S	LI	T
0366	09	79	FD	65		#9	IN	T	HE
036A	OD	A1	8C	D5	OYSTRB	#D	OY	xS	TE R*
036F	08					#8			
0370	CD	B5	21		STREAM	ST	RE	AM	
0373	51	AF	16	7D	EPIT	E.	x	Px	IT
0377	F1	01	15	4F		-	xB	xE	D
037B	04	BA	18	28	F6	28	8F	RYHALL	#4 RO Yx AL HAL L
0382	2F	13	85		GLEN	A	Gx	LE	
0385	9A	F1	4A	18	NOJOY	NO	J	xO	Yx
0389	EF	BF	2D		URAT	U	R	AT	
038C	EF	BF	79		URIN	U	R	IN	
038F	08	2F	C7	13	ASSMAD	#8	A	SI	Gx
0393	91	AF	16	7D	NPIT	N.	x	Px	IT
0397	07	FC	21	8C		#7	S	AY	xS
039B	F2	DD	73		ATTIC	A	TT	IC	
039E	05	15	21	37	MBISAD	#5	Mx	AG	xI C
03A3	05	11	ED	DA		#5	Bx	UT	TO N*
03A8	F7	CF	1D			I	S	@x	
03AB	F6	AE	C5		HOUSE	H	OU	SE	
03AE	FE	C5	F1		USE	U	SE	-	
03B1	07	62	8D	54	HBDRGN	#7	HA	LT	ED
03B5	7F	C5	51		ISEE	I	SE	E-	
03B8	06	11	18	FD		#6	Bx	Yx	T HE
03BD	OE				ADDGMS	#E	(Change to #5 for		
							Scare I Out...)		
03BE	4B	21	3A	9F	ADDRAG	DR	AG	x0	N*
03C2	04	C3	2B	54		#4	SC	AR	ED
03C6	09	AE	DF	11		#9	OU	T	Bx Yx
03CB	04	F5	2D	CF		#4	E	AT	S
03CF	OD	87	DD	85		#D	LI	TT	LE
03D3	0F	F1	18	E5	BLUDEN	#F	B	xL	UE
03D7	F4	BA	16	10	DROP	D	RO	Px	-x
03DB	04				BIRDRM	#4			
03DC	F1	17	B4		BIRD	B	xI	RD	
03DF	FB	AA	15			R	OO	Mx	
03E2	F4	59				D	EN		
03E4	FF	BA	4F		ROD		RO	D	
03E7	32	BB	18	10	CARRY	CA	RR	Yx	-x
03EB	C1	AF	16	7D	SPIT	S.	x	Px	IT
03EF	13	BA	DD	AF	GROTTO	Gx	RO	TT	O*

03F3 12 AB 5C DF	FOREST	Fx OR ES T*
03F7 32 99 AD	CNTMAD	CA NN OT
03FA 6A 17	HOWMAD	HO Wx
03FC F1 9F FF 1C	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 PTRINR 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 EGO file if called with Y=0; is
file at LOCAD if called with Y=1.

1780 B6 41	LOBSCH	LDX EGOLAD,Y	Get pointer to file.
1782 A0 FF		LDY #\$FF	Init object-count.

;

1784 E8	OBFIN	INX	
1785 86 44		STX LOBJAD	Save addr of last obj.
1787 B5 01		LDA 1,X	Set up to test bits
1789 0A		ASL	7 & 6 of each location.
178A 30 F8		BMI OBFIN	b6=1. Not an object.
178C C8		INY	Bump object-count.
178D 90 F5		BCC OBFIN	b7=0. An object. Continue.
178F 60		RTS	b7=1. End of file. Done.

;

```

; 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 #E0CM        Start at End of Cave Map.
1797 B5 00      AOBPLP 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 AOBPLP     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)?
                                BEQ SPATS      Yes, move is OK.
                                CPX #(ADGRM-LNAMAD)  No. At closed
                                |                      grate?
                                BEQ SPCHKD     Yes, disallow Down.
                                CPX #(ADGYM-LNAMAD)  At bridgeless
                                |                      gully?
                                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
;
; LOC SCH (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 LOC SCH STA LOCTNUM 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 LOCTNUM 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 LOC SCH 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 BCIDLH	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 BCIDLH	For LSH, a double
0141 69 01		ADC #1	precision add
0143 85 61		STA BCIDLH	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 BCIDLH	For MSH, a double
0159 69 56		ADC #56	precision add of
015B 85 61		STA BCIDLH	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 BCIDLH	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 BCIDLH	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 BCDSH	For MSH of score, first get upper digit.
0180 4A 4A	LSR LSR	
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 to get segment code.
0188 BD E7 1F	LDA DIGCDD,X	Put MS digit in window.
018B 85 F2	STA WINDO+2	Get next most signif.
018D A5 62	LDA BCDSH	digit and save it.
018F 29 0F	AND #\$F	Was MS digit zero?
0191 AA	TAX	No, so don't blank this digit.
0192 98	TYA	Yes, is this one zero?
0193 D0 03	BNE NOBNK2	Yes -- both zero! Go blank this one too.
	;	Get code for 2nd MSB.
0195 8A	TXA	Fill the remaining slot of the window.
0196 F0 C3	BEQ LZBNK2	
	;	
0198 BD E7 1F	NOBNK2 LDA DIGCDD,X	: CLASS segment. The remaining code
019B 85 F3	LZBNK2 STA WINDO+3	: 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 LOCTNUM	
01A1 C9 06	CMP #6	in the hole
01A3 D0 06	BNE CELSCH	
01A5 A5 3E	LDA BURDEN	
01A7 29 04	AND #4	
01A9 F0 28	BEQ WSLNK	
	;	
01AB A9 00	CELSCH LDA #0	without the rope.
01AD 85 63	STA OBCELR	Go show "F".
01AF 20 03 01	JSR LOCSCH	
01B2 A6 01	LDY #1	
01B4 20 80 17	JSR LOBSCH	
01B7 84 48	STY NOBS	
01B9 F0 15	BEQ CVLINK	Cellar search to see
01B8 B4 00	ORCSSET LDY 0,X	if any treasures
01BD A5 F3	LDA OBCELR	have been left
01BF 19 4C 00	ORA OBJMSK-1,Y	here...
01C2 85 63	STA OBCELR	
01C4 CA	DEX	Anything here?
01C5 C6 48	DEC NOBS	No, check cave visits.
01C7 D0 F2	BNE OBCESET	Yes, so set up
01C9 A2 39	LDX #\$39	OBCELR which will
01CB OA	ASL	have a bit set for
01CC 30 08	BMI PICLNK	each object left
01CE B0 03	BCS WSLNK	in the cellar.
01DD 4C 12 02	CVLINK JMP CAVIS	
01D3 4C 64 02	WSLNK JMP WINSET	
01D6 4C 00 02	PICLNK JMP PICLNK	

; 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 BCDSH	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 OF		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	VISCLP	LDY #\$12	
0222 98		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 NMNBTS	
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 00		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	ARCVLP	LDA VISTBL,Y	Visited House, Glen,
0256 20 16 01		JSR VISCHK	Slit, Forest, Grate?
0259 30 04		BMI 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		BPL ARCVLP	No, keep checking.
0262 A2 1E		LDX #\$1E	Yes, show Class J.
		;	
0264 86 F0	WINSET	STX WINDO	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 09 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
0OFF	
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	ADDOBJ S/R
17A4	
17A5	SPROC segment
17E3	
17E4	(spare)
17E6	

DEC 1979 RCL

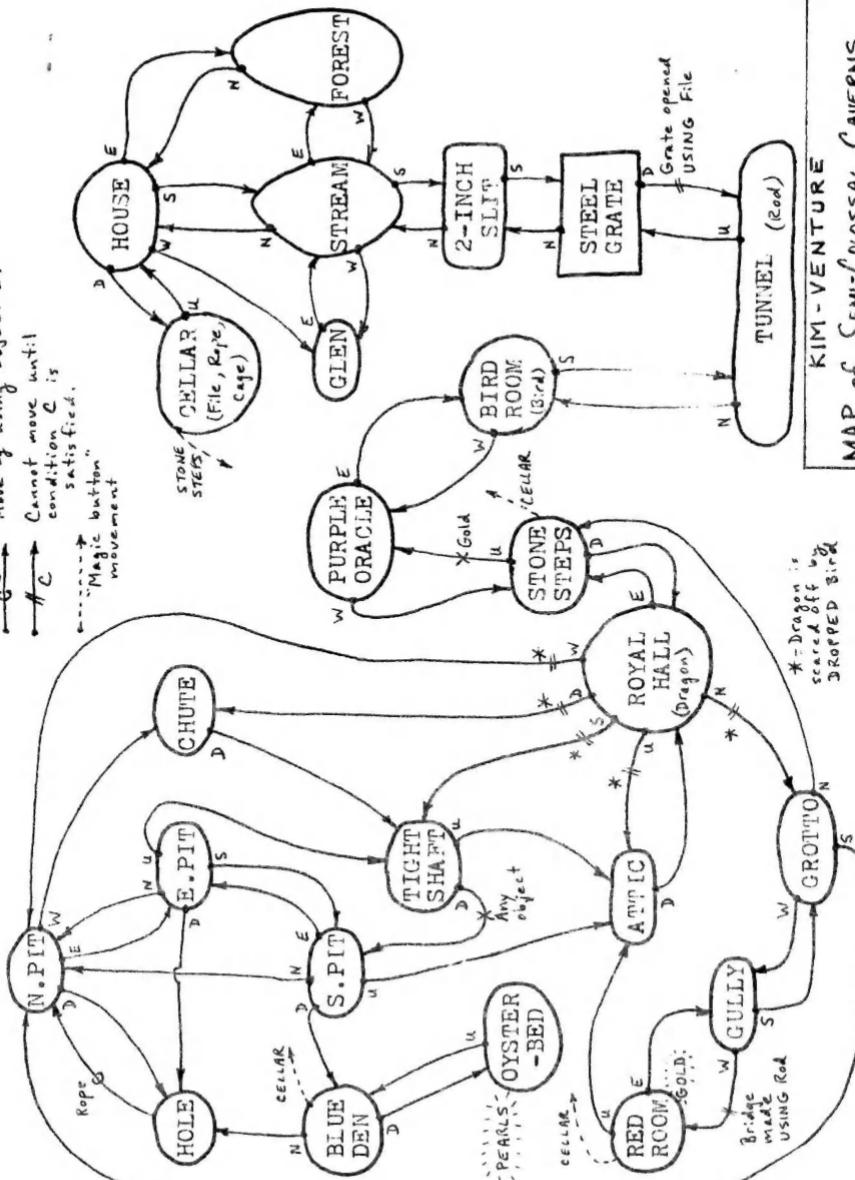
KEY:

A → Cannot move with object A.

B ↔ Move by using object B.

C ↗ Cannot move until condition C is satisfied.

"Magic button" movement



MAP of SEMI-GLOSSY CAVERNS
KIM - VENTURE