



Here is one way of doing this for the vocabulary defined so far:

```

TO START
  MAKE "HERE 1
  MAKE "INVENTORY []
  SET ROOMS
  ASSIGN VARIABLES
  LOOK
  GAME
END

TO GAME
  PRINT1 "COMMAND:
  MAKE "INPUT REQUEST
  IF VALID? :INPUT RUN :INPUT ELSE PRINT (I
DON'T UNDERSTAND)
  GAME
END

TO VALID? :COM
  IF ( ( COUNT :COM ) = 1 ) THEN OUTPUT VAL1?
:COM
  IF ( ( COUNT :COM ) = 2 ) THEN OUTPUT VAL2?
:COM
  OUTPUT "FALSE
END

TO VAL1? :COM
  IF MEMBER? FIRST :COM [INV W E S N LOOK
START] OUTPUT "TRUE
  OUTPUT "FALSE
END

TO VAL2? :COM
  IF ALLOF VALV? FIRST :COM VALN? LAST :COM
  OUTPUT "TRUE
  OUTPUT "FALSE
END

TO VALN? :NOUN
  IF MEMBER? :NOUN [SWORD CHEST SCEPTRE
RING SNAKE] OUTPUT "TRUE
  OUTPUT "FALSE
END

TO VALV? :VERB
  IF MEMBER? :VERB [GET DROP EXAMINE KILL
RUB OPEN] OUTPUT "TRUE
  OUTPUT "FALSE
END

```

THE PROGRAM

You must first enter all the procedures given in the last instalment (see page 775). To begin the game, or to restart it at any time, type START.

```

TO START
  MAKE "HERE 1
  MAKE "INVENTORY []
  SET ROOMS
  ASSIGN VARIABLES
  LOOK
END

```

SET ROOMS sets up the rooms according to the map.

```

TO SET ROOMS

```

```

MAKE "ROOM.1 [[[YOU ARE STANDING AT THE
ENTRANCE] [TO A CAVE]] [] [[E 2]] []]
MAKE "ROOM.2 [[[YOU ARE IN A DARK, DAMP
CAVE]] [] [[S 3] [E 4] [W 1]] []]
MAKE "ROOM.3 [[[YOU ARE IN A DARK, DAMP
CAVE]] [] [[N 2] [E 5]] []]
MAKE "ROOM.4 [[[YOU ARE IN A GREAT
UNDERGROUND CHAMBER]] [] [[N 6] [S 5] [W
2]] [SNAKE.ATTACKS]]
MAKE "ROOM.5 [[[YOU ARE IN A DARK, DAMP
CAVE]] [SWORD] [[N 4] [W 3]] []]
MAKE "ROOM.6 [[[YOU ARE IN A SACRED
SHRINE ROOM] [IN AN ALCOVE IN THE NORTH
WALL] [IS AN ALTAR]] [] [[N 7] [S 4] [E 8]]
[GATE]]
MAKE "ROOM.7 [[[YOU ARE STANDING BY]
[THE ALTAR OF ZOLTOTH THE GILDED] [ABOVE
THE ALTAR IS WRITTEN ] ["LET NO BASE-
METAL APPROACH" ]] [RING] [[S 6]] []]
MAKE "ROOM.8 [[[YOU ARE IN A DARK, DAMP
CAVE]] [] [[S 10] [E 9] [W 6]] [SNAKE.ATTACKS]]
MAKE "ROOM.9 [[[YOU ARE IN A DARK, DAMP
CAVE]] [CHEST] [[S 11] [W 8]] []]
MAKE "ROOM.10 [[[YOU ARE IN A DARK, DAMP
CAVE]] [] [[N 8] [E 11]] []]
MAKE "ROOM.11 [[[YOU ARE IN THE VESTRY
OF] [THE PRIEST OF ZOLTOTH THE GILDED]]
[SCEPTRE] [[N 9] [W 10]] []]

```

END

Logo Flavours

Some versions of MIT LOGO do not have EMPTY?, ITEM, COUNT or MEMBER?. Definitions for these were given in the last two instalments (see page 754 and page 775). In all LCS1 versions, use:

- EMPTYP for EMPTY?
- LISTP for LIST?
- MEMBERP for MEMBER?
- TYPE for PRINT1
- AND for ALLOF
- OR for ANYOF

There is a primitive, EQUALP, which tests whether its two inputs are the same. Use this for comparing lists and words in place of the equals sign (which works for lists on some LCS1 versions, but not on others).

The IF syntax in LCS1 LOGO is demonstrated by this example:

```

IF EMPTYP :CONTENTS [PRINT [NOTHING
SPECIAL]] [PRINT :CONTENTS]

```

The first list after the condition is performed if the condition is true, and the second if it is false.

On Atari LOGO use SE for SENTENCE, RL for REQUEST, and note that ITEM is not implemented.

