THE GRAPHIC ADVENTURE CREATOR

SPECTRUM

INCENTIVE SOFTWARE LTD.

```
==============================
The Graphic Adventure Creator
==============================
```
Document created by Martin M. Pedersen (tusk@daimi.aau.dk) on 20 Jan 97 with
the following comments...

[This text file is made from the original commodore 64 manual to GAC.

  A legal notice. This text and the program is still copyrighted by INCENTIVE
SOFTWARE and you may only use it if you have legally bought a copy of GAC.
]

The following alterations were made by rstuart@ukonline.co.uk on 9th Dec 99:
1. Addition of section 1.2 'Loading Instructions'
2. Re-writing of section 3.2 'Graphics Commands' for Spectrum
3. Addition of appendicies A and B
4. Other minor alterations to make document agree with Spectrum manual
5. Addition of product description from inlay

The Graphic Adventure Creator
Design By: Sean Ellis
Program By: Brendan Kelly
Cover By: Peter Carter
Ransom Pictures: Pete James
Produced By Ian Andrew

THE GRAPHIC ADVENTURE CREATOR

                    For the Spectrum Computer

              CONTENTS :

(1.1) INTRODUCTION
                        ==================

Welcome to the GRAPHIC ADVENTURE CREATOR!

This manual is intended as a simple introduction to writing
adventures using GAC. The GRAPHIC ADVENTURE CREATOR is the best though of as a
small programming language specifically designed for writing adventure games.
Although it is smaller than other programming languages such as Basic or
Pascal it can perform all of the complex tasks needed to write an adventure
program, and although the instructions may seem complicated at first, a little
time spent studying this manual will enable you to write your very own large
scale adventures, far more easily and efficiently than by
using basic or machine code.
    Any adventures that you write using GAC are your own work, and may be sold
as such without any prior permission being sought or payment to us being made.
However in this case, you should include with your adventure something to the
effect that it was written using the GRAPHIC ADVENTURE CREATOR, (C) 1986 by
Sean Ellis/Incentive Software.

                    And now on with GAC!

                    CREDITS

        DESIGN   by   Sean Ellis
        PROGRAMMED   by   Brendan Kelly
        RANSOM PICTURES by   Pete James
        COVER   by   Pete Carter

            Thanks also to Lesley, Giles, Dave and John

The Graphics Adventure Creator (GAC) cassette contains four files.
These are:

       1.   "GAC"   - The Creator itself
       2.   "QS"   - The Quickstart file
       3.   "ADVINMAN" - Adventure in Manual file
       4.   "RANSOM"   - A small Graphic Adventure

To load GAC, type LOAD"" and press enter. Once loaded, QS or
ADVINMAN can be loaded by selecting T from the Main menu - these are
data files.
QS - Quickstart, sets you up with the most commonly used verbs etc
and common messages. A complete listing of its contents is
contained in Appendix B, along with several other useful tables and
charts.
ADVINMAN - Sets up the small adventure in the manual to work through
and even edit!
The fourth file RANSOM is an example of a graphic runnable
adventure created using GAC and can only be loaded from a reset
machine with LOAD"".

          Write an Adventure in 30 Seconds! (For a Bit of Fun!)

First LOAD GAC in
Then "T" to load QS (Quickstart)

Press "R"   - Start Stopwatch!
Press "1"   (enter)
Type "A Cave"   (enter) (enter) (enter)
Break   (Caps-shift, Space)
Press   (enter)   - Stop Stopwatch!

The screen will display:

A Cave
What Now?

Try a few inputs

Presto! - A VERY simple adventure in 30 seconds?
Well look at it this way - it can only get better!

To get back to the main menu press 'Break', 'Break', 'Break'.

This simple example demonstrates how quick and easy GAC is to use.
With a little time and imagination you will soon be creating
professional Graphic Adventures!

                    Let Your Creation Begin!

```
                   (2.1) SECTION TWO - WRITING ADVENTURES
                   ======================================


There are several elements necessary to writing adventures using
GAC. These are displayed on the Main Menu which is the first thing
you will see after loading the program.
   For convenience the menu is arranged in alphabetical order,
which makes it easier for you to remember which keys to press in
order to call up any part of GAC. At the top of the menu screen you
will see the display showing "FREE:" which shows the memory free.
   You may write the elements of your adventure in any order you
like, although you may find that certain orders may make things
easier than others.
   What follows is a brief description of what each element of the
main menu does, together with detailed instructions of how to use
each one. They are presented in an order which you may find
convenient to follow.




                              (2.2) VERBS
                              ===========


Verbs are the words with which you will actually tell the computer
what to do when you are playing an adventure. For example EAT or
DROP. Directions such as WEST or UP are also treated as verbs. GAC
allows you to enter and edit your own verbs when writing an adventure.
Some of the most commonly used verbs are already in the
Quickstart File, however since GAC allows you to enter up to 255
verbs, your imagination is the only limit to the commands you may
incorporate into your adventure.
    After pressing V from the Main Menu, you should see a prompt on
the screen, * EDIT VERBS *, and a pointer >.
    To enter a verb, simply type in a number, then a space followed
by your verb. When you press RETURN the verb will be entered on the
screen in alphabetical order.
If two or more verbs have the same meaning and you wish both to be
accepted, ie., "Get" and "Take", simply allocate to them the same
verb number. Try to think of as many similar words as you can in
this way, since this will make your adventure appear more friendly.


EDITING   - To edit, scroll the list using the cursor keys until the
-------      desired verb is adjacent to use the pointer. Press Return,
             edit the word, press Return again.
DELETE    - Position the word to be deleted by the pointer and press
------       Delete.
RETRIEVE  - Home will retrieve the last word deleted.
--------


    There is a brief summary of these instructions at the bottom of
the screen.


NOTE: The verbs in the Quickstart File may be edited in exactly the
same way as any other verbs.
More information about the use of verbs may be found in section(2.8)
CONDITIONS.

                PRESSING BREAK WILL RETURN YOU TO THE MAIN MENU
```

```
                  (2.3) ROOM DESCRIPTION
                  ======================
```

The term "room" is used to refer to any location in your adventure.
For example, "a forest", "a car", "a strange alien world" and so on.
GAC allows you to use up to 9999 of these rooms, memory permitting,
and to specify the connections to and from each room. You will
probably find it easier if you have decided in advance what each
room is and how the rooms are connected.
    On pressing R from the main menu, the prompt "Which room number
? ...." should appear. At this point enter a room number n, followed
by RETURN which will give the prompt "Room number n is ....". You can
then enter the description which can be up to 255 characters long.
    You will then see the prompt "Connections are.....?". The
connections are specified in terms of a verb, which must part of
the list already specified, followed by a space, followed by the
room number it connects with. For example EAST 20.
    You may have as may connections from a single room as you can
fit into 255 characters, followed by RETURN, i.e.
          EAST 20 WEST 18 NORTH 19 SOUTH 21 JUMP 49 (RETURN)
You will then see a prompt asking for a picture number, which links
this location with the appropriate picture should you wish to
have one. In this case you should enter a picture number followed by
RETURN. If you do not want a picture simple press RETURN. This
returns you to the prompt "Which room number ?..." again.
    All of the above may be edited at any time by using the cursor
keys. BREAK at any point will return you to the WHICH ROOM? prompt,
and BREAK at that point will return you to the Main Menu.

More information about the use of room descriptions can be found in
section (2.7) CONDITIONS.

EDITING TEXT AND NUMBER INPUTS (this applies to all sections)
==
Cursor keys 5 and 8 will move the cursor left and right, DELETE
(Caps-Shift 0) deletes left of the cursor and EDIT (caps-shift 1)
deletes the character at the cursor position and right of the
cursor.


```
                  ===============
                  PANIC BUTTON!!!
                  ===============
```
At any time press BREAK, (several times if necessary) to return to
the main menu.

```
                    (2.4) MESSAGES
                    ==============
```

Messages consist of prompts, descriptions or other comments that
appear on the screen when you play an adventure, i.e. "What now ?",
or "You can't do that". Many if the more common system messages are
contained in the Quickstart file, and are numbered from 240 to 255.
GAC allows you to enter up to 255 messages which may appear at any
point in your adventure.
    Pressing "M" from the main menu will give the prompt "Which
message number ?...". You should then enter a number followed by
RETURN, then type in your message, which may be up to 255 characters
long. Pressing RETURN will enter your message and repeat the "Which
message number ?..." prompt.
    To edit a message once it has been entered, simply recall the
message and edit it using the cursor keys.

    Please note that all message from 240 to 255 should be
defined, since these are essential system message without which
your adventure will not run correctly. You can edit these messages
but you should be careful to preserve the meaning. For example,
message 242, "Pardon ?" could be changed to "Eh ?".

    Pressing BREAK will return you to the prompt EDIT MESSAGE? and
BREAK from there will return you to the Main Menu.

More information about the use of messages can be found in section
(2.7) CONDITIONS.

```
                (2.5) NOUNS AND OBJECTS
                =======================
```

Nouns are the "things" that your adventure will recognise. For
example, BOOK, SWORD and so on.
   GAC frequently requires you to define things both as nouns and
as objects: once as a noun so that the adventure will recognise it,
and then as an object so that the program can move it around and do
things with it. For example, SWORD is defined as a noun, but since
you may wish to pick it up during your adventure, you must also
define it as an object. A further example makes clear another point:
if your adventure requires you to light a lamp. LAMP must be defined
as a noun, and you must also define two objects, AN UNLIT LAMP and A
LIT LAMP .
   When you press N from the main menu, the prompt * EDIT NOUNS *
will appear. Apart from this, the procedure for entering and
editing nouns is exactly the same as that for editing verbs. (2.3)
   When you press 0 from the main menu, the prompt "Which object
number ?...." will appear.
You should then enter a number n from 1 to 255 followed by RETURN.
You will then see the prompt "Object number n is...". You may then
enter the object description consisting of up to 255 characters
followed by RETURN. This will produce the prompt "Starts in room
number...". Enter the number of the room you wish this object to
start in, or press RETURN. Pressing RETURN will assign the object to
room zero, a special room which cannot be travelled to, and which
contains "dead" and "unborn" object. For example, a piece of cake
once eaten will be assigned to room zero.
The prompt "and weighs...." will then appear. Enter the weight you
wish to allocate to this object and press RETURN.
   If the start room number remains at zero, the object is assigned
to ROOM ZERO ; a special room which cannot be travelled to, which
contains "dead" and "unborn" objects. For example, a piece of cake
once eaten will be assigned to room zero.

BREAK will return you to the main menu.

More information about the use of nouns and objects can be found in
section (2.7) CONDITIONS.

```
(2.6) ADVERBS
                =============
```

In the GAC adverbs are used for two sorts of things: firstly to
describe the precise way a verb is used, i.e. "move slowly" or "put
down gently", and secondly they are used to differentiate between
similar nouns. For example, if you adventure has three differently
coloured boxes, the adverbs will specify the colours: RED, GREEN
etc.
   On pressing A from the main menu, the prompt * EDIT ADVERBS *
will appear. Apart from this the procedure for entering and editing
adverbs is exactly the same as that for editing verbs and nouns.

More information on the use of adverbs can be found in section
(2.7) CONDITIONS.

(2.7) CONDITIONS
                    ================

When using GAC the conditions form the internal commands that
simulate decision-making processes within the game. All the terms
you have specified and the data you have entered is now drawn
together to form the structure for your adventure-
    "Conditions" tell the adventure that if certain things are true,
then it is to perform some action, i.e., "if x is true then do y",
though much more complicated conditions may be constructed, i.e.,
    "if x and y are true then do t",
or "if z and q are true then do y and x", and so on.
The GAC has a special format for writing these conditions. For
example, if you have defined verb number 16 to mean "Examine", and
noun number 2 to mean "Room", then a condition using these terms
might read :
    "IF (VERB 16 AND NOUN 2) LOOK WAIT END"
Translated this means if "Examine Room" is typed in, describe the
current room and wait for a new command.
    The Conditional list below separates the words used for the
"Conditions Part", 'If (Verb 16 and Noun 2)' and those used for
the "Action Part", 'LOOK WAIT END'. For subscripts used (a, c, m,
as, o, n, r, v, x and y) - See Appendix B.

i.   CONDITION PART

     VERB v   Is verb v typed ? (v is a number)
                         VERB 7 will give a TRUE answer if verb number
                         7 was typed, otherwise it will give a false
                         answer.
     NOUN n   is noun n typed ? (similar to VERB)
     ADVE a   is adverb a typed ? (similar to VERB)

     HERE o   is object o here ? (i.e. in the same room as
                         you ?)
                         If you are in room 3 and object 1 is also in room
                         3, then HERE 1 will give a TRUE answer because
                         object 1 is here.
     CARR o   Is object o being carried ? (Similar to HERE).
     AVAI o   Is object o available for use ? i.e. here or being
                         carried? (Similar to HERE).
     o IN r   Is object o in room r ? If object 1 is in room 3,
                         as above, then 1 IN 3 will be TRUE, but 1 IN 4
                         will be FALSE.
     WEIG o   Gets the weight of object o. In our adventure,
                         weigh 2 would give the value 20, which is the
                         weight of the rat.

     SET? m   Is marker m set ?
     RES? m   Is marker m reset ?
                         (There are 256 markers, numbered 0 to 255. They
                         are used to store information that can be in one
                         of two states, like doors which are open or shut,
                         light which are on or off, etc. There are three
                         which are important for the adventure program :
                         Mkr 0, if set, means that a room has been
                                 described since it was last reset.
                         Mkr 1, if set, means you are in a light room,
                                 Otherwise you are in a dark room.

Mkr 2, if set, means you have a lamp or some
                             other source of light.
                        If markers 1 and 2 are both reset, then the
                        program will refuse to describe rooms, coming up
                        with the "It's dark" message instead, since you
                        are in a dark room without a lamp.
                        Mkr 3, if set, disables the scoring mechanism
                             when you exit from the game.

     CTR c   Gives you the value of counter c.
     x EQU? c   Is x equal to the value of counter c ?
                        There are 128 counters, numbered from 0 to 127.
                        They are most frequently used to store the number
                        of moves since a particular event, (e.g., moves in
                        the dark). Counter 0 holds the score and counters
                        126 and 127 hold the turns count since the start
                        of the game.
     TURN   Gives the number of turns since the start of the
                        game.

     ROOM   Gives you the room number of the room you're
                        currently in.
     AT r   Gives a TRUE answer if you are in room number r.

     condition AND condition will give a TRUE answer if both of the
                        conditions give TRUE answers.
                        e.g. VERB 1 AND NOUN 2 will TRUE only if verb 1
                        and noun 2 are both typed.
     condition OR condition will give TRUE if either condition is
                        TRUE, or both.
     condition XOR condition will give TRUE if one of the conditions
                        is TRUE and the other FALSE.
     NOT condition will give a TRUE answer if the condition was
                        FALSE, and vice versa.
                        So NOT VERB 1 will be TRUE if verb 1 is NOT typed.

     x < y   Gives TRUE if number x is less than number y
     x > y   Gives TRUE if x is greater than y.
     x = y   Gives TRUE if x is EQUAL to y.

     RAND x   Gives a random number between 0 and (x-1). So
                        RAND 10 will give any number at random in the
                        range 0 to 9.

     VBNO   Gets the number of the verb in this command.
     NO1   Gets the number of the first noun in this command,
                        if any.
     NO2   Gets the number of the second noun in this
                        command, if any. These are used to check word
                        order, and to GET and DROP object without having
                        to have a condition for each one.

  ii. ACTION PART

     LOOK   Describe the room you are in at the moment.
     DESC r   Describe room number r.

     PICT   Turns the picture display on.
     TEXT   Turns the picture display off, giving a text
                        adventure.

```
GET o              Get object number o. If it isn't here, or you've
                   already got it, or it's too heavy then the
                   appropriate message is displayed.
DROP o             Drop object number o. If you haven't got it, then
                   the appropriate message is displayed.
x SWAP y           Exchange objects x and y. In our little
                   adventure, 1 SWAP 5 will exchange the lamp with
                   the lit lamp.
OBJ o              Describe object number o.
LIST r             List all the objects in room number r.
LIST WITH          List all the objects carried with you.
o TO r             Move object number o to room r. To destroy an
                   object move it to room zero.

SET m              Set marker m
RESE m             Reset marker m. See SET? and RES? for an
                   explanation of markers.
x CSET c           Set x to be the value of counter number c.
INCR c             Increase counter c by one. The maximum is 255.
DECR c             Decrease counter c by one. The minimum is 0.
                   Trying to increase past 255 or decrease past 0 is
                   ignored.

GOTO r             Go to room r and describe the new room
CONN v             This checks through the connection table for a
                   connection from the current room using verb v. If
                   one is found, this returns the room number of the
                   room you would have move to if you were to take the
                   connection, otherwise it gives zero. As an
                   example if you were in room 3 in our adventure,
                   CONN 4 would return the value 2, because in the
                   connections from room 3, verb 4 (WEST) would take
                   you to room 2.
STRE x             Set the maximum weight you can carry to x. STRE
                   is short for STRENGTH.
BRIN o             Brings object o here, if it exists.
FIND o             Find objects o and move to it, if it exists.
                   This does not acknowledge any restrictions, (such
                   as connections), so it could be useful in a magic
                   spell.
SAVE               Saves the current game position to tape or disc.
LOAD               Loads the current game position from tape or disc.
                   These are useful for continuing a game after tea
                   (or coffee - the kettle's probably boiled by now!)

WAIT               Waits for a new command.
OKAY               Prints "Okay" and waits for a new command.
EXIT               Stops the game. Since the player is NOT asked,
                   this is recommended for use when the player wins
                   or is killed.
QUIT               The player is first asked if he wants to quit. If
                   he types "Y", the game terminates, otherwise the
                   game continues. On abandoning the game, both the
                   score and the number of moves taken are displayed,
                   assuming this function has not been disabled by
                   setting marker three.
```

```
    MESS ms              Print message number ms.
    PRIN x               Print number x.
    LF                   Prints a Linefeed. Everything from hereon is
                         printed on a new line.
    WITH                 Is equal to the room number of where things are
                         put when you are carrying them.
    HOLD x               Halts the game for x fiftieths of a second, or
                         until a key is pressed. For example, to freeze
                         for 10 seconds, do HOLD 500.
    x + y                As you may expect, returns the value of x added to
                         y.
    x - y                And this gives the value of x - y.
```

There are several things to note when writing conditions. Firstly
all operations are performed strictly from left to right. Secondly
GAC requires spaces both inside and outside brackets, and between
words such as 'verb' and 'noun' and their numbers.
Thirdly, x and y need not be simply numbers, they can be more
complex expressions. However you should note that since GAC
performs all its operations from left to right, you should be
careful about the order in which you enter things. Put any
comparisons first, i.e. if (VERB > 5) AND NOUN 9) QUIT END
   The conditions are checked by GAC at three different points,
and are entered as High Priority, Low Priority and Local.
The flow chart at the back of the manual indicates the order in
which these are actioned.
  The GAC checks for HIGH PRIORITY conditions before prompting
for a player command. It checks for such things as whether you are
still "alive", whether a light a flashing, etc.
Pressing H from the main menu will return the prompt "Which line
number ?...". You should enter these lines in order, ie. 1, 2, 3
etc.
The GAC will not recognise a line number such as '5' when you have
only entered two conditions so far, although it will insert a
condition earlier in your list and remember accordingly. RETURN will
give you the prompt "Line number n is....", at which point you
should enter your condition and press RETURN which will give you the
BREAK returns you to the main menu.

   LOCAL CONDITIONS are checked for after the player has entered a
command, and deal with things local to a particular room, for
example, whether a particular object is in that room. Alternatively
they might execute a player command to go to another room, or open a
door at that location.
   Pressing C from the main menu will give the prompt "Which room
number ? ....". Entering a number followed by RETURN will give the
prompt "Which line number ?....". Thereafter the procedure is
exactly the same as for High Priority Conditions, except that each
Room has it's own set of line numbers, (1, 2, 3 etc).

   LOW PRIORITY CONDITIONS are also checked for after a single
player command is entered, but are not associated with any room. For
example, they might check whether you are carrying a particular
object, independent of the location. After pressing L from the main
menu, the procedure for entering and editing Low Priority Conditions
is exactly the same as that for High Priority Conditions.

   Further examples of High, Local and Low Priority Conditions can
be found in Appendix A.

```
                    (2.8) BEGIN WHERE?
                    ==================
```

The last element of your adventure, at least as far as the text is
concerned, is to specify which room your adventure starts in.
   Pressing B from the main menu will give the prompt to enter the
start location room number.

Entering a room number followed by RETURN means that your adventure
will start at this room. To change the start room, simply enter a
new number.
BREAK will return you to the main menu.


```
                    (2.9) SAVE AND LOAD
                    ===================
```


SAVING

After pressing S from the main menu, you will be asked if you wish
to save a "Data File" or a "Runnable Adventure".
If you press 'D' for 'Data File', the computer will prompt for a
file name. Enter your file name followed by RETURN. This will save
your data which can be loaded into GAC at a later date for further
development.
   If you press 'R' for runnable adventure followed by 'Enter' your
data will be saved as a Runnable Adventure, which can no longer be
loaded into GAC for editing but will run on its own.

LOADING

Pressing T on the main menu followed by a file name will load a Data
file into GAC for editing.

To load your Runnable Adventure type LOAD"" and press ENTER.

NOTE : Do not attempt to load a runnable adventure into GAC. It
will run on its own !


```
                    (2.10) DELETE DATA
                    ==================
```

Pressing X from the main menu will delete all your data.
N.B. Loading a new data file will have the same effect.

```
                 (3.1) SECTION THREE - GRAPHICS
                 =============================


The Graphics part of GAC will enable you to draw pictures to go with
the room descriptions in the text part of your adventure.
   Press G from the Main Menu to enter the Graphics sections. You
will see displayed a high-resolution Window and a Graphics Menu,
together with prompt "Which picture number   ?....".
Enter the number of the picture you wish to design.


The Status Board from left to right:


PEN AND PAPER - The Pen starts "off" the Paper. The cursor will not
-------------   leave a trail and is free to be positioned without
                affecting your picture. The Pen will be in the down
                position whilst actually drawing. (L, E & R)
INK           - The colour in this box shows the current status of
---             the Ink of the Pen.
PAPER         - The colour here shows the current status of the
-----           background, (the paper colour).
X & Y         - These indicators show the current position of the
-----           cursor. The bottom left corner X = 0 and Y = 0.
LAST          - This shows the last command used.
----
PIC           - The picture number you are presently designing.
---


The fill Routine is designed to be fast and it can cope with
irregular shapes, but you may need to press "F" more than once. If
an area is left unfilled, then you can just move the cursor into
that area and press "F" again.
   Here is a simple method which will allow you to get the best
results from your filling. If your shape has a high point, position
the cursor directly under it. If it has a low point, position your
cursor directly under it. Here is an example using a triangle...




                  /^\                 /|\
                 /   \               /-|-\
Only this part  /|----\             /--|--\      All filled
filled         /-+-----\           /---+---\  + = cursor position
              /--|------\         /----|----\
             <---|------->        <-----|----->

                CASE A             CASE B



The area filled is worked out by the fill routine checking up and
down from the cursor position until it encounters a boundary.
It is possible to fill the same area more than once using shading
patterns consisting of different colours.
   Finally, we have another very useful feature. This is the
ability to merge pictures together. For instance, you find that, in
your adventure you have a single basic cave shape, which you want to
use in many of your pictures, but details differ. It is possible to
include a picture already created into the picture you're in, by
pressing "M", typing the number of the picture you want to include,
and pressing ENTER. This is then drawn over the top of anything
already there, (I usually do this right at the start and then draw
around the merged picture). It is a very useful feature which saves
a lot of time and memory too.
```
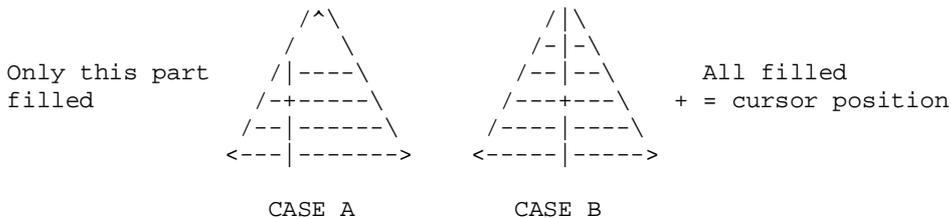
If you should run out of memory in one particular picture, you can get round this by moving to an unused picture and merging the full picture into the new one before continuing with your drawing.

Designing and drawing pictures on the Spectrum is hindered by the Spectrum's own 'Attribute Problem'. If you hold 'G' whilst at a picture, you will see a grid of bright and not so bright squares light up - each square is 8 by 8 pixels. The Spectrum itself will only allow two colours in any square at any time, one is called the Paper Colour and the other the Ink Colour.

By careful positioning of lines, ie. on the edge or corners of Attribute Squares, you can still get several areas of colour joining at one point. As you draw lines, GAC will update all the attributes as the line is elastic banded, this will aid effective positioning. Also as previously mentioned holding the 'G' key will overlay the Attribute Grid over the picture.

There is a demonstration picture (No. 99) on the 'Advinman' file that you can step through, edit, add to, delete or whatever! (Load in Advinman using the T option, press G and enter 99).

By careful design you can create stunning graphics like those included in the Ransom adventure.

                    (3.2) GRAPHICS COMMANDS
                    =======================

Arrow Keys (5, 6, 7 & 8) - Move the cursor
Arrow Keys + Caps. Shift - Move the cursor (steps of 8) quicker.
        L (1st press) - Line - Fixes one end of the line, (pen
                             goes down). Now move the cursor about.
        L (2nd press) - Line - Fixes other end of the line, (pen
                             goes up.
        E (1st press) - Ellipse/Circle - Sets centre (now move
                             cursors).
        E (2nd press) - Ellipse/Circle - Fixes Ellipse/Circle
        R (1st press) - Rectangle - Sets a corner (now move
                             cursor).
        R (2nd press) - Rectangle - Fixes Rectangle.
        D   - Dot - Prints a Dot at cursor position
        I   - Ink Colour
        P   - Paper Colour
        V   - Flash 0 = off 1 = on 8 = transparent
        B   - Bright 0 = off 1 = on 8 = transparent
        T   - Text Area and Border Colour
        G   - Grid showing Attributes positions
        F   - Fills from cursor area in INK
        A   - Fills from cursor area in PAPER
        S   - Shaded fills from cursor in a shade of
                             Current Ink and Current Paper
        W   - Draws whole picture
        M   - Merge picture - prompts for a picture
                             number and merges that picture with the
                             one currently displayed. Note some
                             shades will not merge well and a 'fill'
                             will not fill past obstacles in the
                             present picture.
        C   - Cursor - when held will highlight the
                             cursor position.
Arrow Keys (5, 6, 7 & 8)
and Symbol Shift   - Steps through your picture to allow
                             editing.
S Shift 5 (left arrow) - Steps back one command

```
S Shift 8 (right arrow)  -  Steps forward one command
S Shift 6 (down arrow)   -  Steps back five commands
S Shift 7 (up arrow)     -  Steps forward five commands
NB. When stepping through your picture you can still add and change
the commands as normal.
            Z            -  Jumps to start of picture
Delete                   -  Deletes last command
Caps-shift 9             -  Deletes from current position to end
BREAK 1st time           -  Prompts new picture number
BREAK 2nd time           -  Return to main menu
```

NB. For a Spectrum + or Spectrum 128 the arrow keys move the cursor
in steps of 8, for accurate positioning use keys 5, 6, 7 and 8.
(Left, down, up and right).

Editing pictures

```
,             - Moves back one graphic command.
.             - Moves forward one graphic command.
<             - Moves back 8 graphics commands.
>             - Moves forward 8 graphic commands.

SHIFT *    - Deletes all picture information held after the
               edit position.
Z          - Takes you back to the start of the picture.
f7         - Toggle the graphics cursor speed, between 1 pixel
               per key press (default) and 4 pixels per key press.
BREAK      - Takes you back to the Main Menu.
```

Whilst stepping through the graphic commands the last command will
be highlighted in green. If the last graphic command was a line,
"Dot" will be highlighted in light blue.

                    (4.1) SECTION FOUR - TEST ADVENTURE
                    ===================================

To test your adventure, get back to the main menu and simply press
"ENTER" to enter the adventure.
When are testing and adventure, the way to return to the main
menu is to press BREAK as usual. This gives you the message "Press D
for diagnostics or BREAK to escape....". If you press the "D" key, then
all the markers and counters will be displayed on the screen. For
the markers, a solid box indicates "SET" and an empty one indicates
"RESET". The values of the counters are given as you would expect.
To get back to the game press RETURN. If you press BREAK again, then
you will get back to the main menu.
    The error routine prints out a message indicating the nature of
the error, plus a line of conditions if the error occurred in one of
the condition tables.

"Message not found" means you have referred to a message that does
not exist. If this appears immediately, check that you have included
the system's special messages, numbers 240 and above.

"Room not found" means you have tried to describe or move to a room
that doesn't exist.

"Object not found" means you have tried to pick up, drop or describe
an object that does not exist.

"Marker not found" means you have tried to access a counter which

does not exist.

"Counter not found" means you have tried to access a counter which does not exist.

"Illegal value" means you have tried to look for a verb, noun or adverb with a number greater than 255, or you have tried to load a counter with a number greater than 255.

Here are a few hints for solving errors :
  i.   Suspect typing error. Check that you really mean what you have said.
  ii.   Have you forgotten to enter the room/object/message being referred to? if so enter it.
  iii. Have you forgotten to delete a condition which you didn't need, having deleted the objects/messages used by it?

```
                    (4.2) PLAYER COMMANDS
                    ====================
```

This section deals with the commands a player gives when he plays
the adventure. It is intended to give an idea of the complexity of
the commands the adventure will understand.
   Each command line consists of one or more simple commands,
which in turn consist of a verb, and maybe an adverb and one or two
nouns. Any word which the command interpreter does not understand
as being a noun, verb or adverb is ignored.
Let us take a typical line and see how the program looks at it:

     Get the gold, examine it, put it in the box then go north

Let us assume that the vocabulary includes :
```
                                            ADVERBS: 1 IN
NOUNS :   3 GOLD        VERBS :   1 NORTH
          7 BOX                   7 GET
        255 IT                    8 PUT   16 EXAMINE
```

   The command line is split into separate commands by the
following : ".", ",", "?", ":", "and" and "then", so it
becomes ;

```
     Get the gold      ","
     Examine it        ","
     Put it in the box "then"
     go north
```

Each of these is canned for verbs, adverbs and nouns in that order,
and nay found have their number stored;

```
                        VERB      ADVE      NOUN1      NOUN2
     Get the gold       7         0         3          0
     ---     ----

     Examine it         16        0         255        0
     ------- --

     Put it in the box  8         1         255        7
     --- -- --      ---

     Go north           1         0         0          0
       -----
```

Then all occurrences of noun 255 ("It"), are replaced by the last
noun typed before that, giving:

```
                        VERB      ADVE      NOUN1      NOUN2
     Get the gold       7         0         3          0
     Examine it         16        0         3          0
     Put it in the box  8         1         3          7
     Go north           1         0         0          0
```

These values are the passed to the connection table, then to the
conditions. See the diagram in Appendix B.
Note that all the letters in the vocabulary entries are significant
- there is no truncation to only four or five letters. Thus RIVER
and RIVET, TROUT and TROUSERS, and other like these are
differentiated.

```
                    APPENDIX A
                    ==========
      Definitions and Conditions for "ADVINMAN"


i.NOUNS                 ii.VERBS
-------                 --------
1 TORCH   5 SERPENT   1 NORTH     9 LOOK        15 LOAD
1 LAMP    6 DOOR      2 SOUTH    10 LIST        15 RESTORE
2 RAT   255 IT        3 EAST     10 INVENTORY   16 EXAMINE
3 KEY                 4 WEST     11 QUIT        17 LIGHT
4 GOLD                5 UP       12 TEXT        17 ON
4 BAR                 6 DOWN     12 WORDS       18 EXTINGUISH
4 TREASURE            7 GET      13 PICTURES    18 OFF
5 SNAKE               7 TAKE     13 GRAPHICS    19 UNLOCK
                      8 DROP     14 SAVE        20 SCORE
                      1 N   2 S  3 E   4 W      21 EAT
                      5 U   6 D


iii. ROOM DESCRIPTION
---------------------
Room No. Description (as it would appear)       Connections

1.       You are above the ground.  There is a      (press ENTER)
         cave entrance to the east.
2.       You are in a large cavern.  Passages        E 3
         lead east, west and south.
3.       You are in a cave.  A snake is asleep        W 3
         in a corner and exits lead east and west.
4.       You are by a small lake.  The only exit     (press ENTER)
         is west.
5.       You are outside a castle.  A tunnel leads   (press ENTER)
         to the north and a large door can be seen
         to the east.
6.       You are in the castle strongroom.  A door   W 5
         stands open to the west.


iv. OBJECTS
-----------
No.     Description        Starts in room      Weight

1.      a lamp                   1             10
2.      a dead rat               5             20
3.      a key                    4             1
4.      a gold bar               6             100
5.      a lit lamp               0             100


v. MESSAGES
-----------
   240  What now ?....
   241  You can't
   242  Pardon?
   243  Press a key for another game....
   244  Are you sure ( Y or N ) ?
   245  You've already got that.
   246  You haven't got that.
```

```
   247  You can't see that.
   248  You're carrying too much to pick that up.
   249  Your score was
   250  and you took
   251  It's dark.  You can't see a thing.
   252  I can't find that anywhere.
   253  You can also see.
   254  Okay
   255  turns
```

i. Conditions - Local Don't type in the comments !

 Room 1
    IF (VERB 3) RESE 1 GOTO 2 WAIT END
      If you typed "EAST", reset the dark/light marker, goto room 2
      and wait for a new command.
 Room 2
    IF (VERB 4) SET 1 GOTO 1 WAIT END
      If you typed "WEST", set the dark/light marker, goto room 1 and
      wait for a new command.
    IF (VERB 2) SET 1 GOTO 5 WAIT END
      And a similar condition for going south too.
 Room 3
    IF (VERB 7 AND NOUN 5) MESS 14 HOLD 200 EXIT END
      If you typed "GET SNAKE" then print message fourteen to say
      how it reacts, freeze for four seconds and end the game.
    IF (VERB 3) SET 1 GOTO 4 WAIT END
      Again another move-from-dark-room-to-light-room condition, this
      time east.
 Room 4
    IF (VERB 4) RESE 1 DOTO 3 WAIT END
      I'll ley you work this one out (hint - look at rooms 1, 2, 3)
    IF (VERB 7 AND NOUN 3) CTR 0 + 20 CSET 0 END
      If you typed "GET KEY" then add twenty to counter 0 (the score)
    IF (VERB 8 AND NOUN 3) CTR 0 - 20 CSET 0 END
      If you typed "DROP KEY" then subtract 20 from ctr 0 (the score)
 Room 5
    IF (VERB 3 AND SET? 3) GOTO 6 WAIT END
      If you typed "EAST" and marker 3 is set, (i.e. if the door has
      been opened), goto room 6 and wait for a new command.
    IF (VERB 3) MESS 7 WAIT END
      If you typed "EAST" then print message 7 and wait for a new
      command. Note that the door cannot be opened since if it was,
      the last line would have worked and we would be waiting for a
      new command by now.
    IF (VERB 19 AND NOUN 6 AND CARR 3) SET 3 MESS 10 WAIT END
      If you typed "UNLOCK DOOR" then set marker 3, (mark the door as
      open), print message 10 and wait for a new command.
    IF (VERB 1) RESE 1 GOTO 2 WAIT END
      It is another light-to-dark movement, this time north to room 2

ii. Conditions - Low Priority

Those marked "*" are included in the Quickstart file.

    IF (VERB 20) MESS 249 PRIN CTR 0 MESS 250 PRIN TURN MESS 255
    WAIT END
      If you typed "SCORE", print message 249, your score, message
      250, the number of turns you has and message 255. Then wait
      for a new command.

```
    IF ( NO1 = 0 AND VERB 7 ) MESS 19 WAIT END
      If you typed "GET" by itself or with an unrecognised word,
      print message 19 "I'm sorry, but I don't know what one of those
      is..."
    IF ( NO1 = 0 AND VERB 8 ) MESS 19 WAIT END
      And similarly for "DROP".
    IF ( NO1 = 0 AND VERB 16 ) MESS 18 WAIT END
      And again for "EXAMINE".
    IF ( VERB 7 AND NOUN 1 AND HERE 5 ) GET 5 OKAY END
      If you typed "GET LAMP" and there is a lit lamp here, then get
      it, print "OKAY" and wait for a new command.
    IF ( NO1 < 5 AND VERB 7 ) GET NO1 OKAY END.
      If you typed "GET" and a noun with a number less than 5, then
      get the object with that noun's number. This only works
      because the objects and the nouns which refer to them have the
      same number. (A very useful trick!)
    IF ( VERB 8 AND NOUN 1 AND CARR 5 ) DROP 5 OKAY END
      If you typed "DROP LAMP" and you've got a lit lamp, then drop
      it, print "Okay" and wait for a new command.
    IF ( NO1 < 5 AND VERB 8 ) DROP NO1 OKAY END
      If you typed "DROP" and a noun whose number is less than 5,
      then drop the object with that noun number.
    IF ( VERB 16 AND NOUN 1 AND AVAI 5 ) MESS 5 WAIT END
      If you typed "EXAMINE LAMP" and you have a lit lamp available
      then print out its more detailed description and wait for a new
      command.
    IF ( NO1 < 5 AND VERB 16 AND AVAI NO1 ) MESS NO1 WAIT END
      If you typed "EXAMINE" and a noun whose number is less than 5,
      then print out the message with that noun number, it being the
      more detailed description of that abject, then wait for a new
      command.
  * IF ( VERB 11 ) QUIT END
      If you typed "QUIT" then quit.
  * IF ( VERB 9 ) LOOK WAIT END
      If you typed "LOOK" then describe this room and wait for a new
      command.
    IF ( VERB 21 AND NOUN 2 AND CARR 2 ) DROP 2 2 TO 0 MESS 17 WAIT
    END
      If you typed "EAT RAT" and you are carrying it, drop it, move
      to room 0, thus destroying it, print message 17 ("yum, yum")
      and wait for a new command.
  * IF ( VERB 10 ) MESS 239 LIST WITH END
      If you typed "INVENTORY" then list the objects with you and
      wait for a new command.
  * IF ( VERB 13 ) PICT OKAY END
      If you typed "PICTURES" then turn them on, print "OKAY" and
      wait for a new command.
  * IF ( VERB 12 ) TEXT OKAY END
      And similarly for "TEXT".
    IF ( VERB 17 AND NOUN 1 AND AVAI 1 ) 1 SWAP 5 CTR 0 + 20 CSET 0
    MESS 15 SET 2 WAIT END
      type this all on one line. If you typed "LIGHT LAMP" and you
      have an unlit lamp, then exchange the lit lamp for the unlit
      one, tell the user that he has lit the lamp, increase the score
      by 20 and await a new command.
    IF ( VERB 18 AND NOUN 1 AND AVAI 5 ) 1 SWAP 5 CTR 0 - 20 CSET 0
    MESS 16 WAIT END
      And similarly for "LAMP OFF" ("EXTINGUISH LAMP").
```

```
   * IF ( VERB 14 ) SAVE OKAY END
       If you typed "SAVE" then save the game position to tape or
       disc.
   * IF( VERB 15 ) LOAD LOOK WAIT END
       If you typed "LOAD" then load in a previously saved game
       position.


iii. Conditions - High Priority
-------------------------------
    IF ( RES ? 6 ) SET 6 STRE 111 3 CSET 1 END
       If marker 6 is reset (if this is the first move) then set
       marker 6 to say that it isn't the first move any more, set the
       strength and counter one to hold value three.
    IF ( RES ? 1 AND RES ? 2 ) DECR 1 END
       If you are in total darkness, then decrease counter number one.
    IF ( 1 EQU ? 1 AND RES ? 1 AND RES ? 1 ) MESS 20 END
       If counter one has reached value 1 then print that you can hear
       footsteps.
    IF ( 0 EQU ? 1 ) MESS 21 EXIT END
       If counter 1 has reached zero then tell the player he has been
       "got" by the spider, and end the game.
    IF ( AT 3 AND RES ? 4 AND CARR 2 ) SET 4 MESS 13 DROP 2 2 TO 0
    WAIT END
       If you are in room three with the rat and the snake hasn't been
       fed yet, mark the snake as fed, print the message to say so,
       and move the rat to room 0, destroying it.
    IF ( AT 3 AND RES ? 4 ) MESS 8 EXIT END
       If you are in toom three without the rat and the snake hasn't
       been fed yet, then say that it kills you and exit from the
       game.
    IF ( AT 1 AND CARR 4 ) MESS 9 EXIT END
       If you are back at the start carrying the gold then you win !


GET AND DROP
------------
Here is another useful feature! To save putting a Get and Drop
condition in for each and every object, here are two very useful
condition lines that will enable you to get and drop any object, (up
to No.10 in our example), at any location. To do this simply ensure
the Noun numbers match the Object numbers. ie.

    NOUN 1 = Hat        OBJECT 1 = a hat

and enter Low Priority Condition lines :

    IF ( NO1 < 10 AND VERB 7 ) GET NO1 OKAY END
    IF ( NO1 < 10 AND VERB 8 ) DROP NO1 OKAY END
```

THE QUICKSTART DATA FILE

The "Quickstart" data file contains all the system messages, many
useful verbs, and several of the common low priority conditions
associated with them. This is to allow you to get straight in to
writing the adventure that you want to write, without having to
worry about things that are included in all adventures.
The full contents are :

```
   VERBS                NOUNS
   -----                -----
   6   D                255    IT
   6   DOWN
   8   DROP             MESSAGES
   3   E                --------
   3   EAST             239    You are carrying
  16   EXAMINE          240    What now ?....
   7   GET              241    You can't.
  13   GRAPHICS         242    Pardon ?
  10   INVENTORY        243    Press a key for another game...
   9   L                244    Are you sure ? (Y/N)...
  10   LIST             245    You've already got that.
  15   LOAD             246    You haven't got that.
   9   LOOK             247    You can't see that.
   1   N                248    You're carrying too much to pick that
   1   NORTH                   up.
  13   PICTURES         249    Your score was
  11   QUIT             250    and you took
  15   RESTORE          251    It is dark.  You can't see.
   2   S                252    I can't find that anywhere.
  14   SAVE             253    You can also see
   2   SOUTH            254    Okay
   7   TAKE             255    turns
  12   EXIT
   5   U
   5   UP
   4   W
   4   WEST
  12   WORDS
```

LOW PRIORITY CONDITIONS

IF (VERB 9) LOOK WAIT END
  If you typed "LOOK", redescribe the room you're in and wait for a
  new command.
IF (VERB 10) MESS 239 LIST WITH WAIT END
  If you typed "INVENTORY" then print "You are carrying" and list
  all the objects that are with you.
IF (VERB 11) QUIT OKAY END
  If you typed "QUIT" then ask the adventurer if he is sure, and if
  he responds Y (for YES) then quit, otherwise print OKAY and wait
  for a new command.
IF (VERB 12) TEXT OKAY END
  If you typed "TEXT" then turn the pictures off.

```
IF ( VERB 13 ) PICT OKAY END
  If you typed "PICTURES" then turn the pictures on.
IF ( VERB 14 ) SAVE OKAY END
  If you typed "SAVE" then save the game position to tape or disc.
IF ( VERB 15 ) LOAD LOOK WAIT END
  If you typed "LOAD" then load a previously saved game position
  from tape or disc, then describe the room you are in.
```

i. ACTUAL GRAPHICS COLOURS

```
    0  BLACK
    1  BLUE
    2  RED
    3  MAGENTA
    4  GREEN
    5  CYAN
    6  YELLOW
    7  WHITE
    8  TRANSPARENT - ie. If paper is set to 8, you can draw over all
       colours of paper leaving the paper colour unchanged, (Ink,
       Flash or Bright).
    9  CONTRAST - ie. If ink is set to 9, the ink used will be
       either White or Black, whichever is the greatest contrast to
       the current paper colour. (Or Paper).
```

ii. SUBSCRIPTS USED WITH CONDITION WORDS

```
    a  -  adverb number
    c  -  counter number
    m  -  marker number
   ms  -  message number
    o  -  object number
    n  -  noun number
    r  -  room number
    v  -  verb number
   x,y -  any numbers
```

iii. WORDS USED IN CONDITIONS

```
   Words       Markers      Counters      Rooms       Commands
   -----       -------      --------      -----       --------
   VERB  v     SET   m      CSET    c     GOTO  r     OKAY
   NOUN  n     RESE  m      INCR    c     CONN  r     WAIT
   ADVE  a     SET?  m      DECR    c     ROOM        EXIT
   NO 1        RES?  m      CTR     c     AT    r     QUIT
   NO 2                     x EQU?  c     DESC  r
   VBNO                     TURN          LOOK


   Objects                 Decisions    Tape and Disc
   -------                 ---------    -------------
   GET    o   OBJ   o      IF           SAVE
   DROP   o   LIST  o      END          LOAD
   o SWAP o   WEIG  o
   HERE   o   STRE  o
   CARR   o   o TO  r
   AVAI   o   BRIN  o
   o IN r     FIND  o
```

```
    Other
    -----
MESS  ms                      x = y
x + y                         WITH
x - y                         PICT
condition AND condition       TEXT
condition OR condition        RAND x
condition XOR condition       PRIN x
NOT condition                 HOLD x
x > y                         LF
```

iv. COUNTERS AND MARKERS USED BY THE SYSTEM

```
Counter    0  - Holds the score
           1 x 126 } Count the number of turns since the beginning
         256 x 127 } of the game.

Marker     0  - If set means that a room has been described since
                last reset.
           1  - If set, you are in a light room.
           2  - If set, you have a source of light.
           3  - If set, disables the scoring mechanism.
```

v. RANGES OF NUMBERS

```
               No. allowed        No. of characters

Rooms          1..9999            255
Objects        1..255             255
Messages       1..255             255
Verbs          1..255             35
Nouns          1..255             35
Adverbs        1..255             35
Graphics       1..9999
Markers        0..255
Counters       0..127
[Which score]  0..255
```

```
                          ┌──────────┐
                          │  START   │
                          └──────────┘
                               │
                      ┌────────────────┐
                      │     output     │
                      │ ROOM DESCRIPTION│
                      └────────────────┘
                               │
                        ┌─────────────┐
                        │  TURNS + 1  │
                        └─────────────┘
                               │
                      ┌────────────────┐
                      │ HIGH PRIORITY  │
                      │  CONDITIONS    │
                      └────────────────┘
                               │
                       ╱────────────────╲
240 : "What now?"     ╱     output       ╲
                      ╲  MESSAGE 240      ╱
                       ╲────────────────╱
                               │
                        ╱───────────────╲
                       ╱ PLAYER'S INPUT  ╱
                        ╲───────────────╱
                               │
                      ┌────────────────┐
                      │ CHECK CONNECTION│
                      │     TABLE       │
                      └────────────────┘
                               │
                          ◇ Valid? ◇ ───── Y ──────┐
                               │ N                 │
                                            ┌──────────────┐
                      ┌────────────────┐    │  MOVE TO     │
                      │ LOCAL CONDITIONS│◄── │  NEW ROOM    │
                      └────────────────┘    └──────────────┘
                               │
                          ◇ Valid? ◇ ── Y ──► ◇ New room? ◇ ── Y ──┐
                               │ N                  │ N
                                                ◇ Wait ◇ ── Y ──┐
                                                    │ N
                      ┌────────────────┐
                      │  LOW PRIORITY  │
                      │  CONDITIONS    │
                      └────────────────┘
                               │
                          ( Valid? ) ── Y ──► ◇ New room? ◇ ── Y ──┐
                               │ N                  │ N
                                                ◇ Wait? ◇ ── Y ──┐
                                                    │
                          ( VERB = 0? ) ── Y ──────┘
                               │ N
241 : "You can't"      ┌──────────┐            ┌──────────┐
242 : "Pardon?"        │ MESS 241 │            │ MESS 242 │
                       └──────────┘            └──────────┘
```
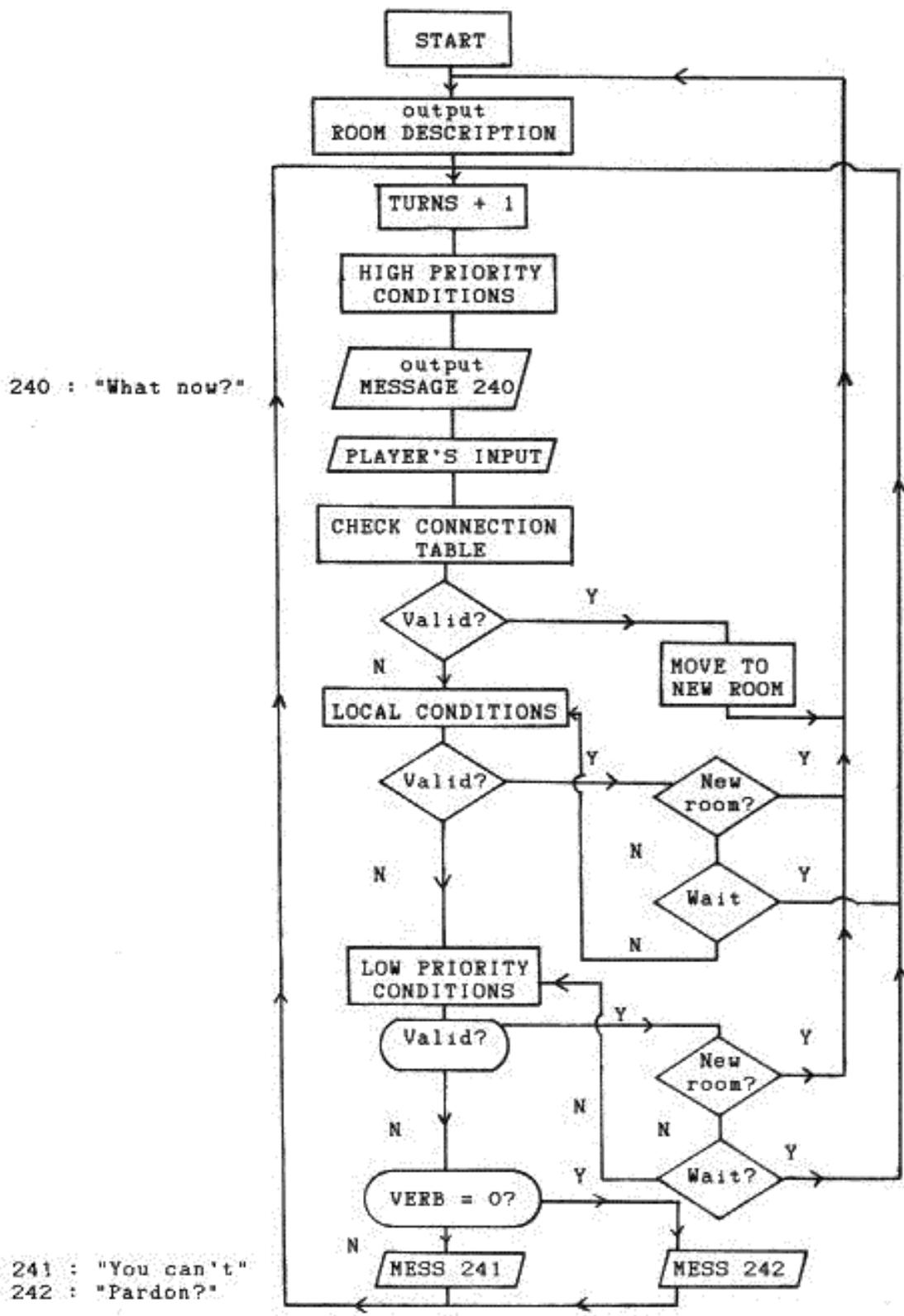
The internal commands QUIT and EXIT return the player to the first
room with the program initialized. There is no end as such.