DUNGEON CRAWL MAP EDITOR
USER'S GUIDE

For the Tandy Color Computer

by Paul Shoemaker

Copyright 2019

INTRODUCTION

Included with the two **Dungeon Crawl** game DSK images is a third DSK image called DCMEDIT.DSK. This disk includes the map editor program which was used to create all the levels in the game. Likewise, you may also use this utility to create your own levels, modify existing levels, or create additional levels to allow your character to venture even deeper into the dark dungeon. The game will automatically react to lower levels by throwing even harder challenges at your character.

However, you must first become familiar with a few important level design principles.

LOADING INSTRUCTIONS

The **Dungeon Crawl Level Editor** will run on a Tandy Color Computer 1, 2 or 3 with a minimum of 64K RAM.

If you have a CoCoSDC or DriveWire configuration, mount the DCMEDIT.DSK image into Drive 0, then type:

RUN "MAPEDIT" and press **[ENTER]**.

If you have a CoCoVGA installed in your CoCo 1 or 2, it is recommended you disable artifact colors by pressing the right-hand CoCoVGA switchboard button five times after the title screen has loaded.

Now let's review the level design principles.

LEVEL DESIGN

Each level is arranged on a 10 x 5 grid containing a maximum of 50 rooms, where each room is itself a 10x10 grid of graphical tiles. Room numbering starts at room 1 in the lower left-hand corner and increments to room 50 in the top right-hand corner, as seen in the illustration below:

| 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
31	32	33	34	35	36	37	38	39	40
21	22	23	24	25	26	27	28	29	30
11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10

It is important to note the relationship of one room to its neighbors. For example, room 25 could possibly connect with the other room numbers like so:

```
          35
           ↑
   24  ←  25  →  26
           ↓
          15
```
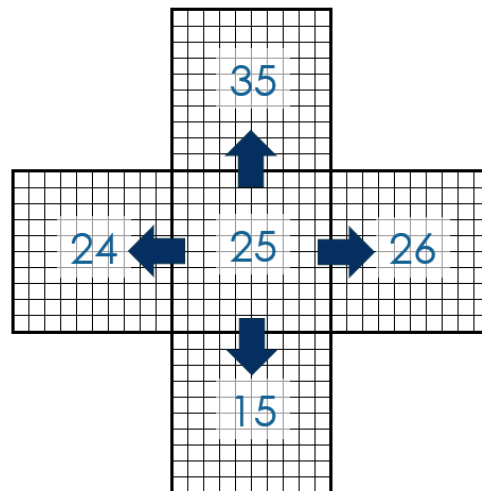
However, every room does not need to connect to its neighbor. It is up to you, the Supreme Level Designer to determine which rooms connect to which of its neighbors. (Don't let all the power go to your head.)

As an example, below is the first level of **Dungeon Crawl** where room 25 in the middle connects only to room 26 to the east and room 35 to the north.
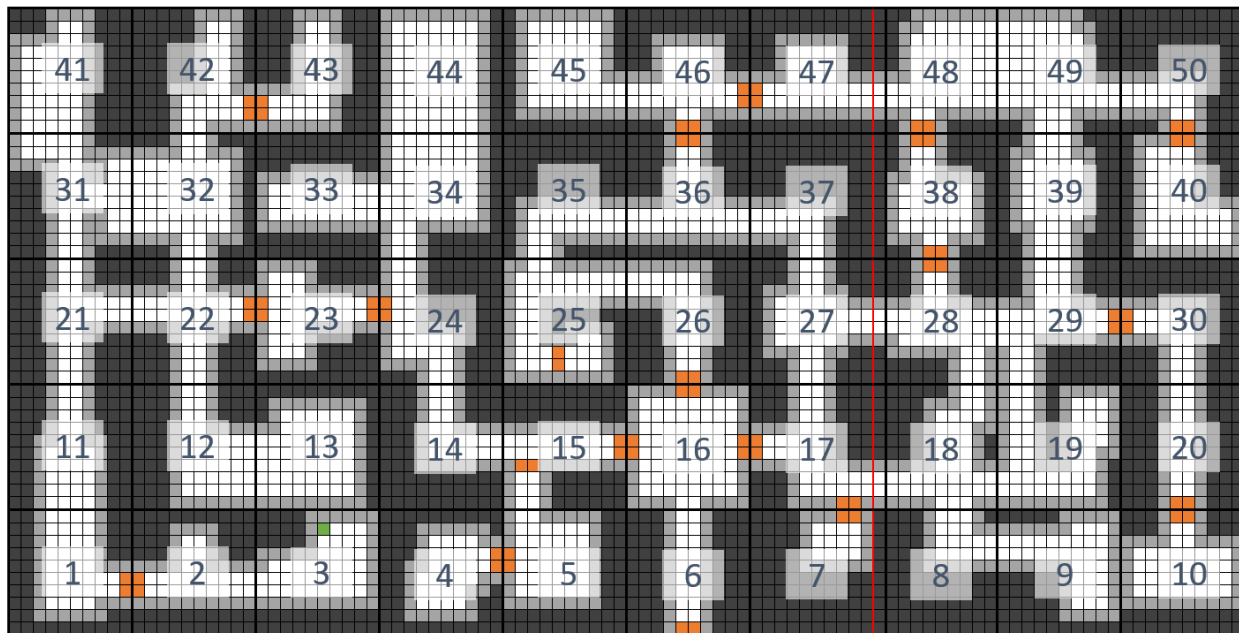
ROOM STRING

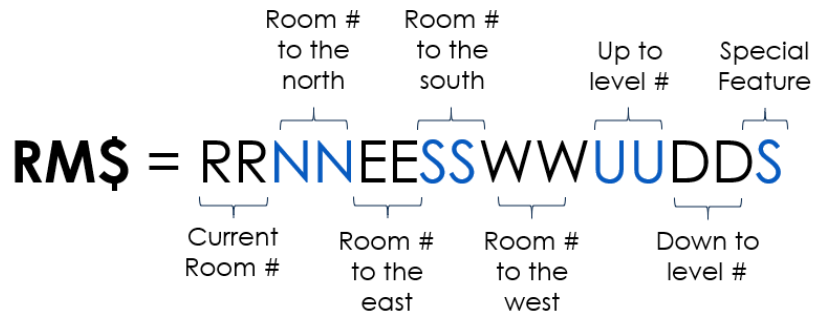In addition to its graphical tile information, every room on every level has a unique room variable string, called the Room String (RM$). This room variable is very important since it defines which rooms connect to other rooms, and which rooms have special features in them such as a throne or a pit.

If each room variable is not specified correctly, the level will not work properly and may cause **Dungeon Crawl** to crash.

RM$ is a 15-character string variable containing 8 separate data elements:

$$RM\$ = RRNNEESSWWUUDDS$$

Room # to the north — NN
Room # to the south — SS
Up to level # — UU
Special Feature — S
Current Room # — RR
Room # to the east — EE
Room # to the west — WW
Down to level # — DD

RM$ must always be exactly 15 character long. Note carefully the number of characters for each data element. The first seven data elements must be 2 characters long; for example, the number one must be represented as "01." Only the eighth data element, controlling whether the room contains a special feature, is a single character.

DATA ELEMENT	PURPOSE	FORMAT
RR	Current room #	**2-digit number** from 01 to 50
NN	Define connecting room to the NORTH of current room #	**2-digit number** 00 = no connecting room in this direction 01 to 50 = connects to that room
EE	Define connecting room to the EAST of current room #	**2-digit number** 00 = no connecting room in this direction 01 to 50 = connects to that room
SS	Define connecting room to the SOUTH of current room #	**2-digit number** 00 = no connecting room in this direction 01 to 50 = connects to that room
WW	Define connecting room to the WEST of current room #	**2-digit number** 00 = no connecting room in this direction 01 to 50 = connects to that room
DD	Define ability to move up one level	**2-digit number** 00 = cannot go up from current room 01 to ?? = can move up to level ##
UU	Define ability to move down one level	**2-digit number** 00 = cannot go down from current room 02 to ?? = can move down to level ##
S	Define whether a special feature exists in the current room. **A room may only have one special feature.** *Ensure the corresponding Special Feature tiles are used when designing the room	**1-digit number** 1 = Pit 2= Teleporter 3 = Altar 4 = Pool 5 = UNUSED 6 = Throne

* There is theoretically no limit to the number of possible dungeon levels, you are constrained only by storage space on the floppy or the floppy image. A standard floppy will hold 13 levels.

To illustrate with an example, let's look at room 40 of the first level of Dungeon Crawl. The only exit in the room is to the north, which connects to room 50. All other directional elements are set as "00" indicating travel in those directions is not possible from this room. Additionally, this room contains the Altar special feature as noted by the number 3 in this data element.


```
RRNNEESSWWUUDDS
RM$:40500000000003
```
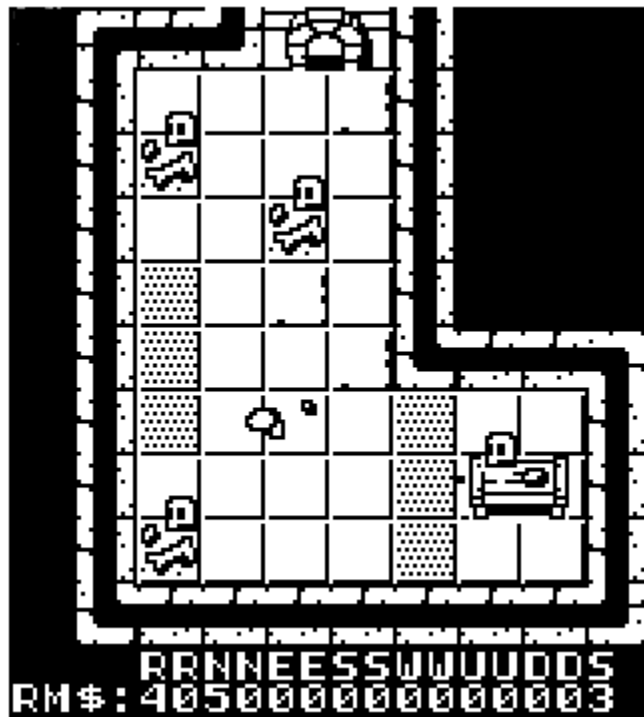
Remember, when placing a Special Feature in a room, in addition to setting the Special Feature data element properly in the RM$ string variable, you must also place the corresponding Special Feature tiles in the room where you want them. In the above example, the Altar is the table with the skull and offering plate on it.

LEVEL FILE NAMING CONVENTION

Each level file must be named as follows:

LEVEL##.DAT

Where ## must be a 2-digit number representing the number of the level. For example, the level one map data file name must be "LEVEL01.DAT".

Failure to follow this naming convention will result in an IE error message as the **Dungeon Crawl** program will not see the level data file name it is expecting.

However, when creating a new level, it is recommended you create the file as "MAP.DAT" (the default used by the **Dungeon Crawl Level Editor**) and when complete, simply rename the file to denote the desired level.

That's it for the fundamentals. Now let's get to the good part.

GETTING STARTED

After running MAPEDIT.BAS, you will be greeted with the following screen:
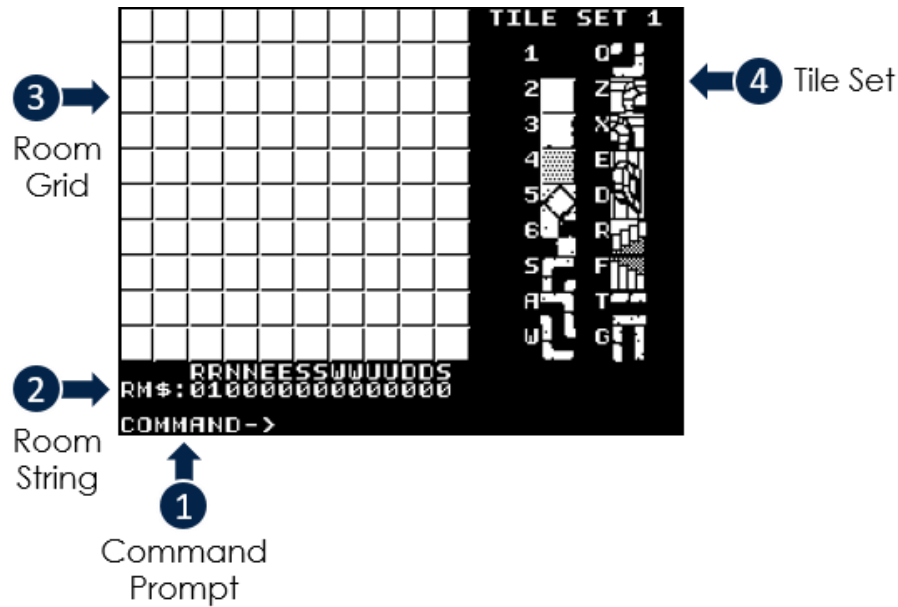
**WELCOME TO THE MAP EDITOR
UTILITY FOR DUNGEON CRAWLER**

**INPUT MAP FILE NAME BELOW
OR PRESS <ENTER> FOR MAP/DAT
?**

Press [ENTER] to proceed creating or updating MAP.DAT (recommended), or type in your own file name following the level file naming convention explained in the previous section.

INTERFACE

The Dungeon Crawl Map Editor interface is divided into 4 sections:

1. **Command Prompt.** Input commands and receive prompts here

2. **Room String (RM$).** Displays the current Room String data elements.

3. **Room Grid.** 10 x 10 grid for graphical tile placement defining the look of the room.

4. **Tile Set.** The current Tile Set, along with the keyboard key assigned to each tile.
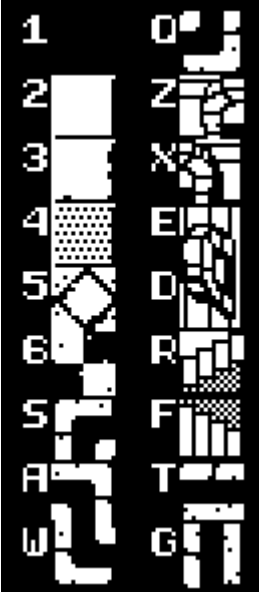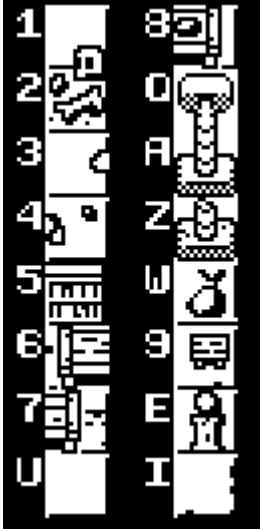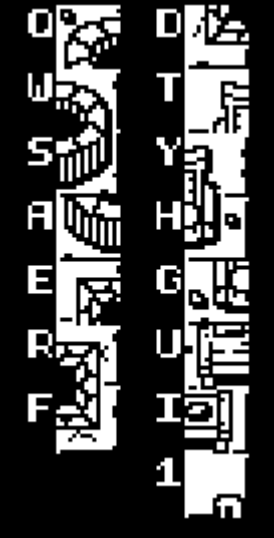
COMMANDS

There are multiple commands used to create, edit and save each individual room of the level.

[ARROW KEYS]	Use the arrow keys on the keyboard to move around the room grid
[SHIFT] + [ARROW KEYS]	Moves to the next adjacent room, loading the current contents of the next room onto the Room Grid. IMPORTANT: Make sure to save the current room before moving, or you may lose changes you've made.
[CLEAR] KEY	Clear the current Room Grid with blank tiles. Since this cannot be undone, you will be prompted to confirm this selection.
[SHIFT] + [CLEAR] KEY	Clear the current Room Grid with white floor tiles. Since this cannot be undone, you will be prompted to confirm this selection.
[SPACEBAR] KEY	Cycles through the three different Tile Sets
[VARIOUS] TILE KEYS	For each Tile Set, the key use to place that tile in the selected cell of the Room Grid is noted. Select the desired cell using the arrow keys, then press the appropriate tile key.
[ENTER] KEY	Saves the room, both the graphical tile placements and the Room String.
[@] KEY	Enters Room String edit mode allowing you to carefully modify this important room information.
[:] KEY	Quit the editor. Make sure to save your current room first.
[?]	Displays the Help Screen to view the commands

TILE SETS

There are three predefined tile sets for use in creating each room of each level of the dungeon. Press the [SPACEBAR] to cycle between the three Tile Sets.

Tile Set #1 Basic Room Architecture Floors, walls, doors, stairs	Tile Set #2 Room Features Furniture, fixtures, debris	Tile Set #1 Special Features Pool, pit, throne, altar

THINGS TO REMEMBER

- **If you are planning to edit the original Dungeon Crawl level maps, make a backup copy of the originals first!**

- Planning out the entire level beforehand on graph paper (or using Microsoft Excel as I did) is recommended.

- Try to make the exits from each room clear. Remember, the only clue the player has about which way he/she can go is by looking at your room.

- Save often during the editing process by pressing [ENTER]. There is nothing worse than losing your edits by moving to the next room to edit without saving your changes to the current room.

- Stairs down on one level should align with the stairs up on the corresponding level.

- The Teleport special feature does not have a corresponding tile. This is by design to make this special "invisible." Zzzap! Surprise!

- A room may only have one special feature, although you may place as many special feature tiles (same or different) as you wish in any room. Whatever special feature you denote in the Room String is the one with which the player will be able to interact.