

*Imagery!*  
A Computerized Role-Playing  
Adventure System  
Adventure Designer's Manual

*Imagery! Adventure Designer*

by

Roy C. Riggs

(c) 1988 Dr. Evil Laboratories  
P.O. Box 190  
St. Paul, IN 47272



## Welcome to the *Imagery! Adventure Designer*

The *Imagery! Adventure Designer* is a very powerful tool that allows you to design your own *Imagery!* adventures-- even if you know virtually nothing about computer programming. The designer is very easy to use as it is completely menu-driven.

This manual will provide you, the prospective adventure designer, with four things:

- 1) A short course on designing your own *Imagery!* adventure
- 2) An introduction to adventure design theory
- 3) An explanation of the features of the designer
- 4) An understanding of the inner workings of an adventure

Designing adventures is a lot of fun. Good luck!

*Imagery! Adventure Designer Manual*

Written by Kent Sullivan

Version 1.0

March 8, 1988

(c) 1988 Dr. Evil Laboratories

## Table of Contents

<b>Getting Started</b> .....	1
What To Do First .....	1
Loading and Running the Designer .....	1
Programs on Your Designer Disk .....	1
<b>Designing an Adventure</b> .....	3
Steps in Adventure Creation .....	3
1: The Idea .....	3
2: Planning .....	3
3: Typing It All In .....	3
Using the Adventure Designer .....	4
1: Preparing a New Disk .....	4
2: Editing the Intro .....	4
3: Searching the Libraries .....	5
4: Adding New Monsters & Items .....	5
5: Creating Rooms & Exits .....	6
6: Placing & Editing Monsters .....	6
7: Placing & Editing Items .....	7
8: Saving the Adventure .....	7
9: Modifying the Adventure Program .....	7
10: Copying Files & Playtesting .....	7
<b>Adventure Design Theory</b> .....	9
Adventure Types & Sizes .....	9
Difficulty Level .....	9
Movement & Friendliness .....	10
Battles .....	10
Magic Spells & Items .....	10
Hints & Tips .....	12
Memory Management .....	12
Mazes .....	12
<b>Intro Edit Program Reference</b> .....	13
Enter Adventure Introduction .....	13
Enter Adventure Name .....	13
Enter Author Name .....	13
Load Intro .....	14
Save Intro .....	14
Exit Program .....	14
<b>Library Editor Program Reference</b> .....	15
Edit Monsters .....	15
Edit Items .....	15
Load Library .....	16
Save Library .....	16

Clear Library .....	16
Dos Commands .....	16
Exit Editor .....	16
<b>Main Designer Program Reference .....</b>	<b>17</b>
Main Menu .....	17
Edit adventure .....	17
Load adventure .....	17
Save adventure .....	18
Load library .....	18
Misc menu .....	18
Exit designer .....	18
Edit Menu .....	18
Goto room .....	18
Edit name .....	18
Edit desc .....	19
Room menu .....	19
Mons menu .....	19
Item menu .....	19
Exit menu .....	19
Main menu .....	20
Edit Monster Menu .....	20
Name .....	20
Desc .....	20
Strn .....	20
Dext .....	20
Frnd .....	20
Cour .....	20
Hitp .....	21
Armr .....	21
Weap .....	21
Reac .....	22
Stat .....	22
Item .....	22
Loca .....	22
Quit .....	22
Edit Item Menu .....	22
Itm Name .....	22
Descrip .....	23
Effect .....	23
Light .....	23
Htpoint .....	23
Value .....	23
Weight .....	23
ItmType .....	24

Locatn .....	24
Quit .....	24
Miscellaneous Menu .....	24
Create adventure .....	24
Clear adventure .....	25
Single/Double drive .....	25
Dos commands .....	25
Main menu .....	25
<b>Appendix A: The Menu Tree .....</b>	<b>26</b>
<b>Appendix B: Adventure Variables &amp; Formulas .....</b>	<b>27</b>
Standard (Non-array) Variables .....	27
Array Variables .....	28
Formulas .....	33
<b>Appendix C: Adventure Program Reference .....</b>	<b>35</b>
Program Overview .....	35
Data Files .....	35
New Commands .....	36
Pseudo-Code Listing .....	36
<b>Appendix D: Flexi-Copy Operation .....</b>	<b>51</b>
<b>Appendix E: Submitting Imagery! Adventures .....</b>	<b>52</b>
<b>Appendix F: Sample Libraries .....</b>	<b>53</b>
Abbreviations .....	53
Medieval Monsters .....	54
Medieval Items .....	54
Modern Monsters .....	55
Modern Items .....	55
Futuristic Monsters .....	56
Futuristic Items .....	56

## Getting Started

### What To Do First

Relax. The designer isn't a hard program to learn to use. You will probably want to read a little of this manual, just enough to get going, and then jump in and let the menus guide you through. There's nothing wrong with that! However, at some point you will want to read this manual in an organized fashion (i.e. when you encounter an unfamiliar option).

You should start reading by finishing the Getting Started section. Then read the Designing an Adventure section, as it breaks down the process of creating an adventure into easy, logical steps. Next, read the Using the Adventure Designer section, which introduces each of the three designer programs. Finally, read carefully the section on Adventure Design Theory. You will need a good understanding of how *Imagery!* is put together in order to use the designer efficiently and to create quality adventures. That should definitely get you going. Later, when you need to look up a particular feature of the designer, you can refer to the reference sections.

### Loading and Running the Designer

The designer is written for the Commodore 64 and 128 (in 64 mode). One Commodore 1541 or 1571 (or compatible) disk drive is required, and a second drive set to device 9 is also useful. To begin, follow these easy steps:

For Commodore 64:

1. Turn on monitor, drive(s), and computer.
2. Insert the designer disk, label side up, into device 8, drive 0 and close the drive door.
3. Type: LOAD "BOOT",8,1 and press **RETURN**.

For Commodore 128:

1. Insert the designer disk, label side up, into device 8, drive 0, and close the drive door.
2. Turn on the monitor, drive(s), and computer. Make sure you are in 40-column mode.
3. The designer will automatically boot on power-up.

NOTE: If your 128 is already on, you may press the **RESET** switch instead of turning everything off then back on.

### Programs on Your Designer Disk

The first thing you will see when you have loaded the designer is a menu with three options. The designer is really three separate programs that work together. The main designer program (option 3) takes up a lot of memory. So, not all of the features we wanted to include would fit into the main program. Two parts of the design process

were separated into their own programs. They are the files "intro editor" (option 1), which allows you to edit the introduction to your adventure (adventure name, author name, & intro text), and "library editor" (option 2), which allows you to create and edit libraries of monsters and items.

The intro editor, library editor, and main designer program all operate under a modified Kernal operating system. This modified Kernal has 4K bytes more free memory for BASIC, which the designer needs. Many other programs, such as *Flexi-Copy*, (described below) won't operate properly under the modified Kernal. Always switch off your computer to reset to the standard Kernal before using these other programs.

Another program on the adventure disk you will use is *Flexi-Copy* (filename "flexi-copy"). *Flexi-Copy* is an all-purpose file copier from Dr. Evil Lab's *Utility Pack I*. You can use *Flexi-Copy* to transfer adventure designer files from one disk to another. *Flexi-Copy* can copy relative files, a file-type many file copiers can't handle-- and a file-type used by *Imagery!*



## Designing an Adventure

This section of the manual will present the steps to take in creating an *Imagery!* adventure and will also provide a walk-through of the designer program.

### Steps in Adventure Creation

There are many ways to go about creating an *Imagery!* adventure-- just as there are many ways to paint a house or assemble a model. As with most things, though, there is one way that involves the least hassle (and the most fun!). After creating quite a few adventures with the designer, those of us here at the Lab have come up with some steps to follow when creating an adventure. You don't have to use these steps, but things will be easier in the long run if you do.

#### 1: The Idea

It's hard to come up with a totally original idea for an adventure. Don't worry, though. Pick a favorite scenario and twist it around to make it unique. If you don't like your idea, throw it away and start over-- you haven't wasted any energy yet. It's better to decide you don't like the idea before you spend a lot of time typing it all in.

Once you have your idea, you can test how good it is by working on the next step, planning.

#### 2: Planning

This part of designing an adventure is the one that everyone seems to hate most, even though it is vital to creating a successful adventure. It takes time to plan an adventure, but it's time well spent. What you plan now won't suddenly surprise you later.

Planning an adventure consists of two major steps: 1) sketching out a map of the adventure, noting where items and monsters should be placed and 2) creating all the monsters, items, and descriptions for the adventure. You generally sketch out the map on graph paper. Many monsters and items you wish to use may already be contained in libraries on the designer disk. If not, you can use the library editor to create and save them for use in your current as well as future adventures.

There are many things to consider when designing an adventure like: 1) How big should it be? 2) How many monsters? 3) What kinds of items should be included? These questions, as well as many other important considerations, are examined in the Adventure Design Theory section.

#### 3: Typing It All In

Once you have your adventure planned out on paper, transferring it to the computer is really the easy part. If you attack this task in an orderly fashion things will go much smoother than if you just haphazardly enter "this" then "that". Think of the designer as an "adventure processor". It's like a word processor but it's for adventures. The designer simplifies the actual coding process-- the arrangement of text and numbers that describe an adventure. (You could design an *Imagery!* adventure with just a simple sequential text file editor and the list of variables from Appendix B, but it would be

very difficult.)

Here is an outline of the steps we use when creating an adventure:

- 1) Prepare a new disk for an adventure
- 2) Enter the intro text, adventure name, and author name
- 3) Search libraries for monsters/items that are needed
- 4) Add any new monsters/items to their appropriate libraries
- 5) Enter all the rooms, their descriptions, and exits
- 6) Retrieve all the monsters from the libraries, modify, and place them
- 7) Retrieve all the items from the libraries, modify, and place them
- 8) Save the adventure
- 9) (Optional: it's advanced) Modify the base adventure program for special effects
- 10) Copy necessary files and playtest the adventure

Let's now take a closer look at each of the ten steps in using the designer.

## Using the Adventure Designer

As mentioned in Getting Started, the designer is really three programs: 1) the intro editor ("intro editor"), 2) the library editor ("library editor"), and 3) the main designer program ("designer"). The first step in using the designer begins with preparing a new disk.

### 1: Preparing a New Disk

Load the main designer program (see Getting Started for instructions). You should see the Designer Main Menu after it's all loaded in.

The designer is composed of a tree of interconnecting menus. Think of the Main Menu as the ground level of the tree (the "root"). From here you can proceed down the tree, "branching" farther and farther away. It would be a good idea to fold out Appendix A: The Menu Tree now so you can get an idea of how the menus are organized.

You can select any of the options by using the **CURSOR UP** and **CURSOR DOWN** keys to move the blue highlight bar. Press **RETURN** to select the highlighted option. You can also use a joystick plugged into Port 2 to move and select.

Select the "Misc menu" option from the Main Menu. Next, select "Create adventure" from the Misc menu. If you have two disk drives, make sure the "Single/Double drive" option is set to "Double drive". Put a blank disk (or one that's ok to erase) into device 8 (if you're using one drive) or device 9 (if you're using two). Press **F7** to prepare the disk for use. The designer will format the disk and create the necessary files for a new adventure.

### 2: Editing the Intro

The intro editor allows you to enter the screenful of text that is the introduction to the adventure. This is the screen you see right after you leave the Towne to go on an adventure. You also set the name of the adventure and the author's name with this

program.

To load the intro editor, see the "Loading and Running the Designer" section. Once loaded, you will then see the Main Menu. Choose the appropriate option to enter or edit the intro screen, the adventure name, or the author name. Note that the intro screen can be a maximum of 24 lines long. The editor won't allow you to enter more.

When typing in the intro, be sure to format the text on the screen just as you want it to appear in the adventure. For example, don't break a word between lines without using a hyphen.

### **3: Searching the Libraries**

Libraries are made up of groups of monsters and items for use in adventures. Each library generally has monsters and items from the same era, as in "medieval" or "futuristic". Libraries can hold a maximum of 15 monsters and 15 items.

When creating a new adventure, you should first check any existing libraries for monsters and items that you need. The designer disk comes with three libraries-- one medieval, one modern, and one futuristic. To search the libraries, first load the library editor. See the "Loading and Running the Designer" section for loading instructions.

Once loaded you will see the Library Main Menu. The library editor is basically the Edit Monster and Edit Item menus from the designer, with a few exceptions. Select "Load library" from the menu by using **CURSOR UP** and **CURSOR DOWN** to move the blue highlight bar. Press **RETURN** to choose the highlighted option. You can also use a joystick plugged into Port 2 to move and select.

Choose the library you wish to search from the list. Once the library has loaded, choose either "Edit monsters" or "Edit items" to look at either monsters or items. Choose the "Edit" option from the menu at the bottom of the screen. Move up/down to choose the monster or item you wish to view.

Once you are familiar with the libraries you can then decide what new monsters and items you need to create for your adventure.

### **4: Adding New Monsters & Items**

You can add new monsters and items to an existing library or create your own new library. Since libraries hold only 15 each of monsters and items, you may eventually have many libraries with monsters and items from the same eras.

To add monsters or items to an existing library, first load that library into the library editor. See the previous section, Searching the Libraries, for loading instructions. On the other hand, if you wish to make a new library, select the "Clear library" option instead.

Next, select either "Edit monsters" or "Edit items" from the Library Main Menu. Next choose the "Create" option (monsters) or the "Design" option (items) from the small menu at the bottom of the screen.

The editor will then make a new monster/item and put you automatically into edit mode. Type in all the relevant info for the monster/item. See the Library Editor

Program Reference section for an explanation of the different features of items and monsters.

Once you have created all the monsters and items you need, make sure you select "Save library" from the Main Menu. You will be prompted with the current library name. You can press **RETURN** to accept the current name. If you enter an empty string for the name, the save will be aborted.

### 5: Creating Rooms & Exits

The next four parts to editing an adventure are all accomplished with the main designer program. Load it now. (See the Loading and Running the Designer section for loading instructions.)

Next, select "Load adventure" from the Main Menu. Make sure your adventure disk is in the appropriate drive. A new adventure has only one room, and it is empty. When the adventure has loaded you are ready to begin.

All rooms in an adventure have a room number. No two rooms will ever have the same number, but the number for a particular room may change as you add and delete rooms from an adventure. To edit a room, select "Edit adventure" from the Main Menu. The Main Menu will be replaced by the Edit Menu. The screenful of information you see corresponds to one room in the adventure. The number in the upper left corner is the room number.

There is a list of options, somewhat like the Main Menu, at the bottom of the screen. You can move up and down as well as left and right here to choose any of the options. You can select "Room menu" to add and delete rooms.

You can edit the name (the short description) or the description (the long one) by selecting "Edit name" or "Edit desc". When you edit the room name, you will also select whether the room is naturally lit or not.

To move from room to room, select "Goto room". You will be presented with a list of room numbers and names to help you identify the rooms. Move up/down to choose the new room.

To edit the number and kinds of exits for the room, select the Exit menu. You will see a new screenful of info. Vertically along the left is the list of possible directions to exit any room (10 ways) and Quit. Setting exits couldn't be easier. Move up/down to select the direction. Press **RETURN** or the fire button to toggle whether the exit is secret or not (secret exits are shown in red). Select a direction. Move right to highlight the room to which the exit will lead. Then use **CURSOR UP** and **CURSOR DOWN** to select the correct room for the exit. It's that easy!

An important thing to remember is that exits are one-way. If you make the north exit from room 1 lead to room 2, the south exit from room 2 will not automatically lead to room 1. This feature gives you greater flexibility, allowing exits to be secret on one side only, or only be one-way. Also, you could edit the north exit from room 1 to lead to room 2 and the east exit of room 2 to lead to room 1, giving the impression that the connecting passage is curved.

When you're all done, select "Quit" to return to the Edit Menu. Your changes to the

exits will be visible here also.

### 6: Placing & Editing Monsters

The first step in adding monsters is to load the library which contains the monsters you wish to use. Only one library may be in memory at a time. Select "Load library" from the Main Menu. You will then be able to choose which library to load. Next, select "Edit adventure". Move into the first room you wish to place a monster by using "Goto room". Then select "Mons menu". You will have a choice of Edit, Summon, Kill, and Quit. Choose "Summon" and select a monster from the library. Once you have summoned a monster, choose "Edit" to edit any of its characteristics. Please see the Main Designer Program Reference section for a synopsis of the different monster traits, as there are too many to go into here.

The "Kill" option allows you to delete a monster from the adventure. Any items the monster is carrying will also be deleted. "Quit" returns you to the Edit Menu.

### 7: Placing & Editing Items

Placing and editing items works much like placing and editing monsters. First, make sure you have the appropriate library loaded. Next, select the room in which you wish to place the item. Then, select "Summon" from the Item menu. Choose the item to place with **CURSOR UP** and **CURSOR DOWN**. You can now go back and edit the item ("Edit" on the Item menu). Note that to place an item on a monster you first place the item in a room then "Edit" the item and select "Locatn" (location). You can then move the item to a monster. As with monsters, the editing options are too numerous to go into here. Please see the Main Designer Program Reference section for a description of the different features you can alter.

"Remove" on the Item menu is to items as "Kill" is to monsters. You can delete any item in a room or on a monster in the adventure.

### 8: Saving the Adventure

Once you have entered and edited the information for an adventure, you then save the adventure. To save, select "Save adventure" from the Main Menu. That's all there is to it!

### 9: Modifying the Adventure Program

The base *Imagery!* adventure program ("adventure" on any adventure disk) has many features that allow a lot of flexibility in designing. Sometimes, though, you may wish to add special effects not covered in the standard features. Often these extras pertain to a specific room or a specific set of conditions. These special cases always require modifying "adventure".

A good knowledge of BASIC and "adventure" is needed to even consider making any modifications. Most designers won't bother with any changes in any of their first adventures-- maybe even never. If you want to try making some changes to "adventure", carefully read the Adventure Design Theory section and Appendices B and C. Some modifications are harder than others to make-- many aren't worth the trouble. Watch the *Designer's Bench* in *The Image* for hints and tips on making modifications

(and other advanced topics).

### 10: Copying Files & Playtesting

After you have saved your adventure you're almost ready to test it! First, though, you need to copy over some files to your adventure disk. Here's what you should do:

- 1) Use the SCRATCH command from BASIC to scratch the following files from your adventure disk:
  - a) boot
  - b) super dos 2
  - c) title
  - d) dump.obj
  - e) adventure
- 2) Load *Flexi-Copy*, the file copier on the designer disk. For a quick run-down of its operation, see Appendix D: *Flexi-Copy* Operation.
- 3) Use *Flexi-Copy* to copy the files A-E above from a finished adventure disk, such as *Beneath Mount Imagery*, to your adventure disk.

That's it! You're ready to try out your adventure. Note that you only have to copy these files once per adventure disk.

Make a note of any descriptions that seem out of whack and any monsters that don't act correctly (too hard to kill, etc.). Also check to see that all items work as planned. A thorough playtesting is usually followed by another editing session-- and then another round of playtesting.

Once you get your adventure into shape, have some friends play it. Other people often catch things you've missed (especially spelling errors and plot inconsistencies). When your adventure is complete, it's time to send it to Imagination so other *Imagery!* players can enjoy it. Please see Appendix E: Submitting *Imagery!* Adventures, for the details. Have fun!

## Adventure Design Theory

In order to create a good *Imagery!* adventure, you as a designer need to have at least some idea of how rooms, monsters, and items all interact in an adventure. As a player you appreciate a smoothly-flowing adventure that “plays right”. This flow comes from the designer understanding the elements of the system with which he is working. A good designer knows how to balance battles with puzzles to make a fun, yet challenging, adventure. This section will present “behind-the-scenes” information that will bring you a greater understanding of *Imagery!* and how to more effectively use the designer. Let's begin with a discussion of the different kinds and sizes of adventures.

### Adventure Types & Sizes

As mentioned in the *Imagery! Player's Manual*, almost any kind of adventure is possible with *Imagery!* There are only three basic adventure types currently supported in the Towne program weapons shop, though: medieval, modern, and futuristic. There is a separate library for each of these genres included with the designer. It's important to understand the different weapons and monsters that are appropriate for each time period.

A little bias is built into *Imagery!*, as you may have noticed— medieval weapons are the only ones affordable by a new player. We did this on purpose. Medieval adventures are the most popular, so we felt that most beginning players and designers would concentrate on them. (We also wanted to discourage players from taking a plasma auto-cannon into *Beneath Mount Imagery* and wasting each of the monsters with one blast!)

*Imagery!*'s main strength is its flexibility— yet this is also its possible downfall. Because of the limited memory of the C-64 and the size of the adventure and the designer programs, adventures can't be as big as we had originally hoped. *Beneath Mount Imagery* has 23 rooms— not the limit, but approaching it. The three components of an adventure— rooms, monsters, & items— are all competing for the available memory. The designer has a ceiling of 50 rooms, 50 monsters, and 50 items. You will most likely never reach this limit because of limited memory. Keep in mind that the more monsters and items you have in an adventure, the fewer rooms you will be able to squeeze in.

So, how do you fit everything in? The best advice we can give is to keep things in about the same proportions: around 20 each of rooms, monsters, and items. Then, massage things around to include everything you can.

### Difficulty Level

When playing an adventure, nothing is more frustrating than running into a monster which can't be beaten or a puzzle that can't be solved. As a designer, you need to understand how to make things challenging but not impossible. A good understanding of the guidelines for parameters describing various monsters and items is vital. Look over Appendix F: Sample Libraries for a run-down of many of the common monsters and items and how they have been defined. Plan your monsters and items to fit in with the existing ones. Make sure you don't make things too easy as well!

Going hand-in-hand with this understanding is the practice of always testing your adventures with a character of the level which is likely to attempt it. The ultimate failure occurs when no one wants to play your adventure but you-- and by testing it with an average character with average knowledge and powers you will ensure that never happens.

### **Movement & Friendliness**

When exploring an adventure, a character is free to move from room to room, picking up items and searching for secret exits, as long as there are no unfriendly monsters around.

When an adventurer encounters a monster, the monster's reaction to the player is determined. The first turn after the monster and player have met, the player can leave the room or launch a first attack, but not search the room or pick up objects. During the second turn, the monster's reaction is determined.

Two characteristics make up a monster's reaction to the player: its reaction rating and its friendliness (a percentage). The reaction rating is the monster's overall attitude toward the player. The friendliness percentage reflects the likelihood of a monster's reaction changing after meeting a player. If a monster's reaction is determined to be unfavorable, the monster will attack the player. Please see Appendix B: Adventure Variables & Formulas, for a detailed description of reaction determination.

### **Battles**

When the adventurer is in battle, he must either befriend, defeat, or scare away any unfriendly monsters before valuable items can be taken. The adventurer also can only leave the room by the exit which he entered. Therefore, time spent in battle is important to the successful completion of an adventure.

Battles can be divided into five types of action:

- 1) Attacks by the adventurer (and any friendly monsters)
- 2) Attacks by unfriendly monsters
- 3) Effects of special/magical items that help the adventurer
- 4) Effects of special/magical items that harm the adventurer
- 5) Added (non-standard) special programming effects (not common)

The effects of weapons and armour are pretty obvious-- weapons injure monsters and characters while armour absorbs damage from weapons. The effects of magical spells and items aren't quite as obvious, though.

### **Magic Spells & Items**

There are four standard magic spells which an adventurer can learn: Speed, Heal, Blast, and Power. Heal can be used on the adventurer or a monster. Speed and Power can



only be used on the adventurer. Blast only effects monsters. Here's what they do:

- 1) Speed- doubles player's dexterity for 0-25 turns (random value).
- 2) Heal- restores 0-20 points of health (random). If used on a fleeing monster, it will stop fleeing.
- 3) Blast- does 0-10 (random) times adventurer's level points worth of damage to a monster.
- 4) Power- has one of 6 outcomes (10% chance each)
  - a) cursed for 25 turns
  - b) protected for 25 turns
  - c) regenerating for 25 turns
  - d) poisoned for 25 turns
  - e) invisible for 10 turns
  - f) asleep for 10 turns

The remainder of the time (40%) nothing happens. See below for an explanation of the effects of the Power spell.

Magic items are more flexible generally than spells because both the number of times the power can be used and the duration (if applicable) can be set. Also, magic items only effect the player, never monsters. A magic item can have one of nine powers, six of which correspond to one of the Power spell's six abilities. Two others are kin to the Speed and Heal spells. The nine powers are:

- 1) cursing
- 2) protection
- 3) regeneration
- 4) poisoning
- 5) invisibility
- 6) sleeping
- 7) speed
- 8) healing
- 9) teleportation

Cursing increases the chance of being hit in battle. Protection is the opposite, as it decreases the chance of being hit. Regeneration restores one point of health per turn, while poison depletes one point per turn. Adventurers who are invisible can't be seen by monsters, while those who are asleep are defenseless-- they can't move or attack. Speed works just like the Speed spell (dexterity times 2). Healing restores all health points. Teleportation transports the adventurer instantly to a random room in the adventure.

Duration doesn't apply to teleportation or healing as their effects are instantaneous. Set durations for those items to 0 in the designer.

NOTE: Although a monster can be tied up or asleep, it has to be set that way with the designer-- the player can't do either of those things to it. On the other hand, a monster is capable of going berserk or fleeing in battle, depending on how courageous it is and

the amount of damage a player does to it.

## Hints & Tips

A few other things should be considered as you design your adventures. Let's first talk a little more about memory management— what things take up memory and what things don't.

### Memory Management

All of the short descriptions (called the names) for rooms are held in memory at all times. The same applies for the attack phrases for natural weapons and the effects for magic items. These should therefore be as short and concise as possible. On the other hand, the long descriptions of rooms, items, and monsters are read in from disk when required and displayed directly on the screen, taking no memory. These, then, can be quite descriptive. Even though the short descriptions need to be short, try to distinguish between like monsters, items, and rooms to avoid confusion, unless you need very similar groups. One situation where this can be effective is discussed next.

### Mazes

One of the neat tricks you can pull with room descriptions involves creating a maze. If you make the short and long descriptions exactly alike group-wise for the rooms in a maze, it will be more difficult to map. See rooms 15-18 of *Beneath Mount Imagery* for an example. Adventurers won't be able to tell easily when they have entered another room since everything will look the same. We'll be covering these and other neat techniques in each issue of *The Image*, so stay tuned!

## Intro Edit Program Reference

The Intro Editor Main Menu has six options, each of which is fairly self-explanatory. The Intro Editor has three functions: 1) editing the text introduction to an adventure, 2) giving the adventure a name, and 3) setting the author name.

### Enter Adventure Introduction

Choose this option to create or edit an adventure's introduction (the screen of text presented when you first go on an adventure from the Towne). You will be able to move around freely on the screen with the cursor keys to edit the intro-- somewhat like a mini word processor.

The intro can be a maximum of 24 lines long. The editor won't let you enter text on the 25th line. Also, you can't place any text in the last (40th) column of a line. Other than that, you are free to place text anywhere on the screen. Make sure you format the intro exactly as you would like for it look within the adventure-- hyphenate words in the correct places, etc.

Generally, an intro will have the first line blank, with the second line being the name of the adventure (centered), and then two more blank lines before the start of the adventure description. Here's a familiar example:

```

                <Blank line>
            Beneath Mount Imagery
                <Blank line>
                <Blank line>
    Lately there have been whispered
    rumors around the towne. Many years (etc.)
  
```

When you have finished entering text, press **Esc** to return to the Main Menu. If you want to save any changes you have made, reply with a "y" to the "Save changes?" question. Otherwise, none of your work will be saved.

### Enter Adventure Name

Choose this option to edit the name of your adventure. The current name will be displayed ("**<adventure>**" if just created) and you can then edit it. Adventure names should describe some aspect of the adventure. When you are through editing, press **RETURN** to go back to the Main Menu.

### Enter Author Name

Choose this option to set the adventure author's name. The current author will be displayed ("**<your name>**" if just created) and then you can edit it. When you are

through editing, press **RETURN** to go back to the Main Menu.

### **Load Intro**

Use this option to load in the intro information for an adventure. Make sure you put your adventure disk in device 8, then press **RETURN**. The editor will then load in the intro information.

If you have just created an adventure disk with the designer, there won't be an introduction for the adventure. The name of the adventure will be set to "<adventure>" and the author to "<your name>".

### **Save Intro**

Use this option to save the intro information for an adventure. Make sure you have your adventure disk in device 8, then press **RETURN**. The editor will then save the revised intro information.

### **Exit Program**

Choose this option to exit the intro editor. Make sure you have saved your work before exiting! If you chose this option by mistake, answer "n" to the "Are you sure?" prompt.

NOTE: To restart the editor without erasing the information in memory, use "GOTO20", not "RUN".

## Library Editor Program Reference

The Library Editor is really a duplication of the Edit Monster and Edit Item menus from the main designer program. The purpose of the Library Editor is to provide a way to create and maintain libraries of monsters and items that can be used in many different adventures. Libraries are raw materials for adventure construction: you use what is in them as a base, modifying and changing to fit your needs for a particular adventure.

Because the Library Editor is so much like the main designer program, we won't duplicate the explanation of any functions that are also found in the main designer program. Refer to the discussion of those features in the Main Designer Program Reference section.

### Edit Monsters

As mentioned above, most of the features of this option are identical to the ones found under the Edit Monster Menu in the main designer program. There are some differences, however, that we will discuss here.

- 1) The "Summon" option from the main designer program is not available, obviously. It is replaced by the "Create" option, which allows you to create a new monster from scratch.
- 2) The "Desc" option isn't available in the Library Editor. Each monster should have a unique description which reflects its particular situation, so it is not included in the Library Editor.
- 3) The "Weap" option only allows editing of the monster's natural weapon (if you want it to have one). You can equip monsters once they are placed in an adventure.
- 4) The "Item" option isn't available in the Library Editor. Any items you want a monster to have will be dependent on the adventure, so you should do this once the monster has been placed in the adventure.
- 5) The "Loca" option isn't available in the Library Editor. Monsters in a library obviously can't be placed in particular places since libraries aren't connected to specific adventures.

### Edit Items

As mentioned above, most of the features of this option are identical to the ones found under the Edit Item Menu in the main designer program. There are some differences, however, that we will discuss here.

- 1) The "Summon" option from the main designer program is not available, obviously. It is replaced by the "Design" option, which allows you to create a new item from scratch.
- 2) The "Descrip" option isn't available in the Library Editor. Each item should have a unique description which reflects its particular situation, so it is not included in the Library Editor.
- 3) The "Locatn" option isn't available in the Library Editor. Items in a library obviously can't be placed in particular places since libraries aren't connected to

specific adventures.

### **Load Library**

This feature works exactly like the Load Library function in the main designer program.

### **Save Library**

Use this feature to save any library you create or modify. Names can be a maximum of 13 characters long, as they are prefaced by "lb.". If you are saving a previously-created library, the Library Editor will scratch the old version of the library and replace it with your edited version. Make sure you save libraries onto the designer disk and not an adventure disk!

### **Clear Library**

Choose this option to clear any library in memory so that you can create a new library.

### **Dos Commands**

This option allows you send commands to the disk drive (device 8 only). Note that if you wish to scratch a library from the designer disk, precede the name with "lb.", i.e., to scratch the library "temp", use the name "lb.temp".

### **Exit Editor**

Use this option to exit the Library Editor. Make sure you have saved your work before exiting! If you chose this option by mistake, use "GOTO1000", not "RUN", to re-start and preserve any library in memory.

## Main Designer Program Reference

As discussed in the Preparing a New Disk section, the main designer program is composed of a tree of interconnecting menus. This section will describe each of the designer's functions, one menu at a time. You might want to have Appendix A: The Menu Tree handy to guide you from menu to menu.

The designer has 6 major menus, each with several options. The hierarchy of the menus is:

### Main Menu

- 1) Edit Menu
  - a) Edit Monster Menu
  - b) Edit Item Menu
  - c) Exit Menu
- 2) Miscellaneous Menu

All of the designer's menus wrap around so you can move quickly within them. To select menu options, you can use the up and down cursor keys to move the highlight bar and **RETURN** to select. You can also use a joystick plugged into Port 2.

Any time you are presented with a list of items from which to select, such as a list of monsters to choose from a library, you can move up and down through the list. Choose "<Nobody>" or "<None>" if you selected that particular menu option by mistake. Press **RETURN** to make your selection.

When entering text for any applicable menu option, such as the name for a room, you can move freely within the body of the text with the cursor keys. The editor is always in insert mode as well. Use **DEL** to erase unwanted characters. When you are done editing, press **RETURN** to save the text.

Let's now discuss the menus in the above order, beginning with the Main Menu.

### Main Menu

The Main Menu is the first menu of the designer. From here you can proceed up the menu tree to the rest of the menus and options. Besides the four options and two menus available, the Main Menu also shows the name of the adventure you are currently editing and the name of the library you are currently using. Let's now take a look at each of the options on the Main Menu.

#### Edit adventure

This option leads to the Edit Menu. In order to use this option you must have either just created a new adventure or loaded one from disk.

#### Load adventure

Use this option to load an adventure for editing. Before selecting this option, make sure you have an adventure disk in the appropriate drive. If you have one drive,

replace the designer disk with an adventure disk. If you have two drives, make sure the adventure disk is in device 9.

### **Save adventure**

Use this option to save an adventure you have edited. Before selecting this option, make sure you have an adventure disk in the appropriate drive. If you have one drive, replace the designer disk with an adventure disk. If you have two drives, make sure the adventure disk is in device 9.

### **Load library**

When you select this option, the designer disk will be searched for any libraries. A list will then be presented from which you can select the one you wish to use. Select "<None>" if you chose this option by mistake.

### **Misc menu**

This option leads to the Miscellaneous menu, which is discussed below.

### **Exit designer**

Use this option to leave the designer and return to BASIC. Make sure you have saved the adventure you were editing with the Save adventure option before exiting!

NOTE: If you chose this option by mistake, type "GOTO1000", not "RUN" , to re-start and preserve any adventure in memory.

## **Edit Menu**

This screen presents information about the room currently being edited. The current room number is displayed in the top left corner. You can also see the room's name, whether the room is naturally light or not, the room's description, any monsters and items in the room, and any exits out of the room.

From here you can edit the current room's characteristics, move to a new room, or move to the Monster or Item menus. Here's a closer look at your options:

### **Goto room**

Use this option to select a different room to edit. When you select this option you will be presented with a list of the valid rooms in the adventure. When you move to a new room the display will change to show the new room's information.

### **Edit name**

The room name is the short description of the room. All descriptions are prefaced with the text "You are". See the Hints & Tips section for some tips on creating effective room names. The maximum length of a room name is 71 characters, not counting the "You are".



After you have edited the room name, select whether the room is to be naturally light or dark. Indoor rooms normally will be dark.

#### **Edit desc**

The room description is the long description of the room, read from disk the first time you enter a room or when you type "Look". See the Hints & Tips section for some tips on creating effective room descriptions. The maximum length of a room description is 253 characters.

#### **Room menu**

By selecting this option you may add or delete rooms. Select "Add" to add a room to the end of the room list. You will be moved into the new room automatically.

Select "Delete" to delete the room you are currently in. Note that all items and monsters in the selected room are also deleted. Any exits into the deleted room from other rooms are set to "<0, Nowhere>". Finally, the last room in the list is moved to fill in the gap left by the deleted room. Any exits into it from other rooms are adjusted as well.

Select "Quit" to return to the Edit Menu.

#### **Mons menu**

This option allows you to summon, edit, and kill monsters. "Summon" presents you with a list of the monsters in the current library. Any monsters you choose will be placed in the current room. "Kill" allows you to remove monsters from anywhere in the adventure, not just the current room. Note that any items the monster is carrying will also be deleted. For a description of the "Edit" option, please see the Edit Monster Menu section below.

#### **Item menu**

This option allows you to summon, edit, and remove items. "Summon" presents you with a list of the items in the current library. Any items you choose will be placed in the selected room or on the selected monster. "Remove" allows you to remove items from anywhere in the adventure, not just the current room or current monster. For a description of the "Edit" option, please see the Edit Item Menu section below.

#### **Exit menu**

This option leads to the Exit screen, where you may add, delete and edit the exits for the current room. Vertically along the left is the list of possible directions to exit any room (10 ways) and Quit. Move up and down to the exit direction you wish to change. Press **RETURN** to toggle whether the exit is secret or not. Secret exits are shown in red.

Move right then up/down to change which room that particular direction leads. The room number and an abbreviated description are shown for each exit direction. Use "<0, Nowhere>" for unused exits.

“Quit” returns you to the Edit menu.

### **Main menu**

Selecting this option returns you to the designer main menu. See the Main Menu section above for more information.

### **Edit Monster Menu**

This menu presents information about the monster you have selected to edit. From this menu you can edit all of a monster's characteristics. There are 14 options available. For some sample monsters, please see Appendix F: Sample Libraries. Let's examine now each of the options:

#### **Name**

Adventurers identify monsters by their names, so try to make each name unique. In some cases you might want a group of like monsters, though-- maybe for an ambush. The maximum length of a monster name is 38 characters.

#### **Desc**

The monster description is displayed when an adventurer first meets a monster. It should paint a picture that helps identify the monster and the situation as the adventurer meets it. Maximum length of monster descriptions is 253 characters.

#### **Strn**

A monster's strength determines how well he hits when attacking in battle. 18 is high for a human.

#### **Dext**

A monster's dexterity determines its overall agility and reaction speed in battle. 18 is high for a human.

#### **Frnd**

A monster's friendliness percentage helps determine how well it will like an adventurer. The higher the percentage, the more likely it is to like an adventurer. Friendliness affects the calculation of a monster's reaction to an adventurer. See the “Reac” section for more information.

#### **Cour**

A monster's courage percentage determines how long it will stay and fight when injured. A monster with a courage of 100% will stay and fight to the death, while one

with low courage will flee when hurt.

### **Hitp**

A monster's hit points are a measure of the amount of damage it can take before dying. A monster with 20 hit points can take 20 points of damage. For comparison, a first level adventurer has between 10 and 20 hit points. Also, note that adventurers gain between 2 and 8 hit points per experience level.

### **Armr**

Selecting this option presents you with three options: Natural, Item, and Quit. There are two types of armour for monsters: natural and man-made. Strong natural armour is applicable for only certain monsters (dragons, etc.) but all monsters could have some natural armour.

If you select Natural, enter the armour bonus. The armour bonus is a percentage reflecting the basic strength of the monster's natural armour. The greater the percentage, the better the armour is and more difficult it is to inflict damage on it. An armour bonus of 15% is about maximum for most monsters (with monsters like dragons being exceptions).

If you select Item, next choose which armour the monster is to have from the list of items in the current library. Note that items which aren't armour are also in the list. You will only be able to choose an item that is armour. To edit the armour's characteristics, see the Item menu option below.

Quit returns you to the Edit Monster Menu.

### **Weap**

Selecting this option presents you with three options: Natural, Item, and Quit. There are two types of weapons for monsters: natural and man-made. Dangerous natural weapons are applicable for only certain monsters (ones with claws, fangs, etc.)

If you select Natural, enter the weapon bonus. The weapon bonus is a percentage reflecting the basic strength of the monster's natural weapon. The greater the percentage, the better the weapon is and more likely it is to inflict damage (score a hit) in battle. A weapon bonus of 15% is about maximum for most monsters (those with excellent claws or fangs might be exceptions). Then enter the phrase which will be displayed when the monster attacks. Most phrases begin with a "-ing" verb, like "going for the throat". Maximum length of an attack phrase is 38 characters.

If you select Item, next choose which weapon the monster is to have from the list of items in the current library. Note that items which aren't weapons are also in the list. You will only be able to choose an item that is a weapon. To edit the weapon's characteristics, see the Item menu option below.

Note that the weapon you select, whether natural or man-made, becomes the owner's ready weapon.

Quit returns you to the Edit Monster Menu.

### **Reac**

A monster's reaction determines whether or not a monster will attack an adventurer. A negative number means that the monster is mad and will attack on sight. The greater the number, the more mad it is. A reaction of 0 means the monster is neutral and won't attack. A reaction of 1 means the monster will be friendly and will fight with the adventurer while he is in the current room. A reaction > 1 means the monster will join the player's party and will follow him throughout the adventure. The greater the number the more loyalty the monster has for the player.

### **Stat**

Status reflects which of the six possible states a monster is currently in. You can select either "normal", "asleep", "tied up", "dead", "berserk", or "fleeing". "Normal" means that the monster is free to move and fight. Asleep or tied up monsters can't move or fight-- they are helpless. Fleeing monsters will run away from a fight, while berserk monsters will seek out battle. A berserk monster also has a temporary courage of 100%. Monsters usually have "normal" statuses.

### **Item**

Selecting this option presents you with the Item Menu, which is explained in Main Menu section above. Note that any item summoned will be given to the monster you are editing, not just placed in the current room.

### **Loca**

Choose this option to change the location of the monster you are editing. Select which room to place the monster. If you decide not to move the monster, choose the room you are currently in.

### **Quit**

Select this option to return to the Edit Menu. See the Edit Menu section above for more information.

## **Edit Item Menu**

This screen presents information about the item you have selected to edit. From this menu you can edit all of a item's characteristics. There are 10 options available. For some sample items, please see Appendix F: Sample Libraries. Let's examine now each of the options:

### **Itm Name**

Adventurers identify items by their names. Many items will have very similar names, but special items should have unique names. For example, a magic sword might be

“FireStorm” instead of just “sharp sword”.

### **Descrip**

The item description is displayed when an adventurer Examines it. It should paint a picture that helps identify the item. Maximum length of item descriptions is 253 characters.

### **Effect**

An item's effect is the short description printed when the item is activated. Weapons, normal items, and magic items have effects, while armour and treasure don't. Some sample effects:

- 1) Weapon - “The spiked balls twirl around”
- 2) Normal item - “As you rub the lamp it begins to glow!”
- 3) Magic item - “You slip the ring on your finger and”

The maximum length for an effect is 79 characters. Note that for magic items that the effect should end with “and” so that the result can be added to the end, as in: “You slip the ring on your finger and you are protected.”.

Armour doesn't have an effect because the base adventure program prints “Wearing the <armour>.” whenever you “activate” a piece of armour (i.e., wear it). Treasure doesn't have an effect because money can't be activated.

### **Light**

The value for light determines how many turns an item will give off light before expiring. A value of -1 indicates that the item will give off light forever. Only normal items (such as lamps and torches) and weapons (enchanted) can be lit. All other items (armour, magic, and treasure) can't be lit.

A non-enchanted light source, such as a torch or a lamp, should burn for a limited number of turns, while a magic lamp or sword might provide light forever.

### **Htpoint**

An item's hit point total is a measure of the amount of damage it can take before breaking. Items which give off light will be extinguished when broken. A readied weapon, when broken, will unready itself, while armour, when broken, will automatically be taken off (but the player will still be carrying it).

### **Value**

An item's value is the number of gold pieces it is worth. Items sold to the pawn shop in the Towne never bring full price.

### **Weight**

An item's weight describes how heavy it is and how hard it is to move. Remember that a player can carry a maximum of eight times his strength. Items you don't want the player to be able to take can be assigned a weight of 9999. The base adventure

program will tell players to leave items alone that have this weight.

### **ItemType**

There are five types of items: weapon, armour, treasure, magic, and normal. For a weapon, enter the number of dice of damage the weapon does per attack. Next enter the companion sides per die. The greater the number of sides, the greater the range of damage the weapon can do. For example, a ten-sided die can do between 0-10 points of damage, depending on the roll. The dice of damage determines how many times the die is rolled. The separate rolls are added together to give the final damage. Finally, enter the attack bonus, which is a percentage reflecting the basic strength of the weapon. The greater the percentage, the better the weapon is and more likely it is to inflict damage (score a hit) in battle. An attack bonus of 25% is about maximum for most weapons. Specially enchanted weapons might have a higher attack bonus.

For armour, enter the defense bonus, which is a percentage reflecting the basic strength of the armour. The greater the percentage, the better the armour is and more difficult it is to inflict damage on it. An armour bonus of 25% is about maximum for most armour.

Treasure is automatically converted into money and added to the player's gold when the player successfully finishes the adventure.

For a magic item, select its power from the list. Next, select the duration for the effect, if applicable. Then, enter the number of charges the item has. For a discussion of the different magic powers, please see Magic Spells & Items in the Adventure Design Theory section.

### **Locatn**

Choose this option to set the location of an item. You can choose a room or a monster.

### **Quit**

Select this option to return to the Edit Menu. See the Edit Menu section above for more information.

## **Miscellaneous Menu**

This menu is a collection of features that don't fit anywhere else. Let's take a look at them:

### **Create adventure**

Select this option to prepare a disk for a new adventure. If you have two disk drives, make sure the "Single/Double drive" option is set to "Double drive". Put a blank disk (or one that's ok to erase) into device 8 (if you're using one drive) or device 9 (if you're using two). Press **F7** to prepare the disk for use. The designer will format the disk and

create the necessary files for a new adventure.

**Clear adventure**

Use this option to clear the current adventure in memory but not the current library. Make sure you have saved the adventure first if you want to use it later.

**Single/Double drive**

Use this option to set whether you are using one or two disk drives. Two disk drives are useful because you can use device 8 for the designer disk and device 9 for the adventure disk.

If you have a second drive (device 9) turned on when the designer is first loaded, the program will default to double drive operation.

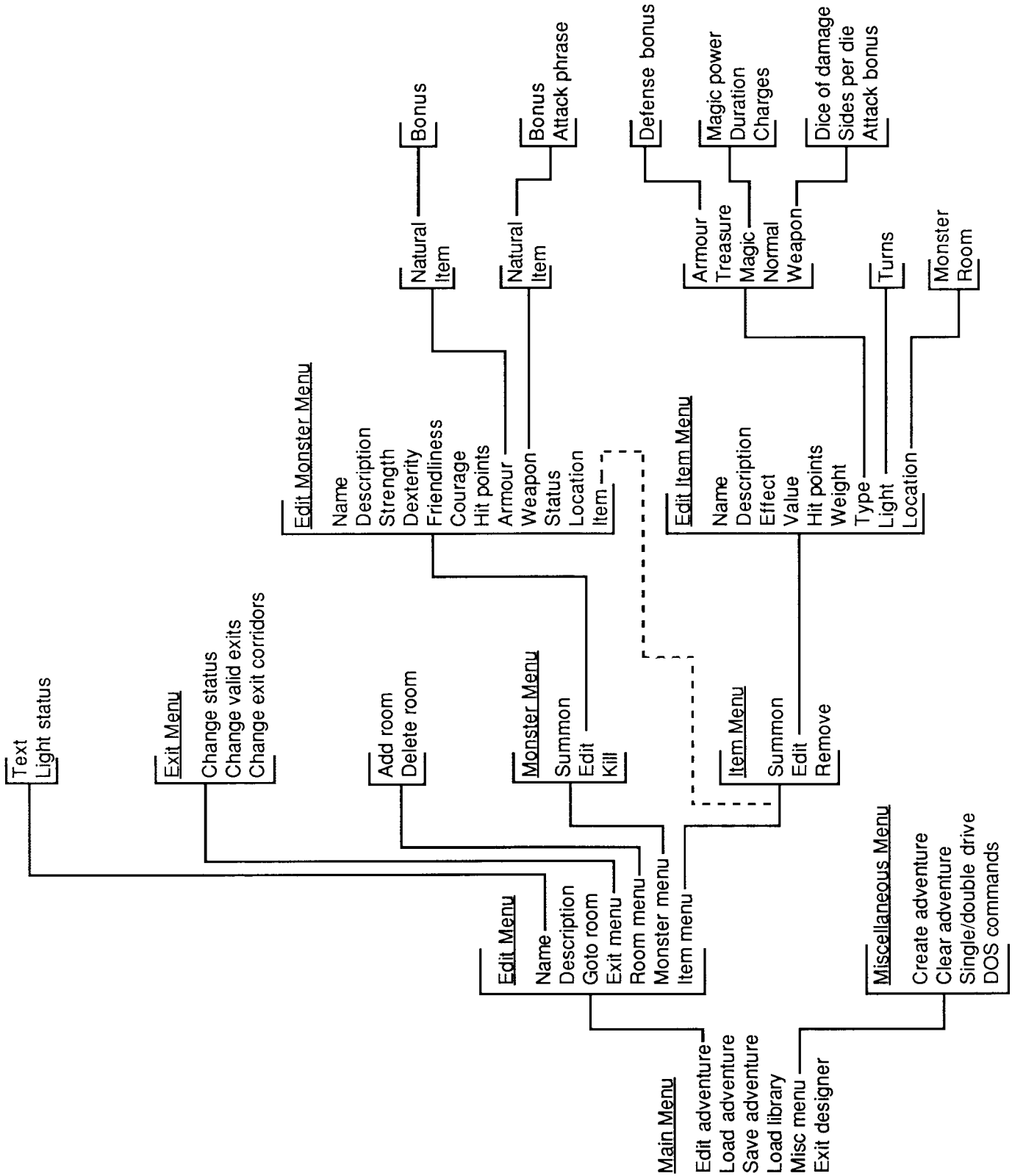
**Dos commands**

Use this option to send commands directly to either disk drive (8 or 9). This feature is useful for scratching and renaming files. To send a command to device 9, preface the command with a "9:", as in "9:s0:myfile" to scratch "myfile".

**Main menu**

Selecting this option returns you to the designer main menu. See the Main Menu section above for more information.

### Appendix A: The Menu Tree





## Appendix B: Adventure Variables & Formulas

The following is a description of all the variables and formulas used in the *Imagery!* base adventure program.

### Standard (Non-array) Variables

a	which item
a\$	keypress
ab	armour bonus
al	allow flag
at	attacker
ba	amount of money player has in bank
bt	battle flag
c\$	player's command
cf	number of friends in battle
cm	number of enemies in battle
cu	curse flag
cz\$	player's last command
d	number of dice of damage
da	defender's armour
df	defender
dm	total damage from attack
dr	direction
e	error code read from drive
e\$	error type read from drive
ex	player's experience
fl	flee flag
fr	friendliness
gl	amount of money player owns
hp	ratio of damage to total hit points
ht	chance to hit / result of attack
in	invisibility flag
j	temporary
kl	kill flag
lt	light flag
lv	player's level
m	monster
na	number of items in adventure
nc	number of legal commands
nm	number of monsters
nr	number of rooms
nz	total number of items (in adventure & carried)
o	last room entered
p	record pointer
po	poison flag
pr	protection flag
r	current room
re	regeneration flag
rl	die roll
s	number of sides on dice
s\$	subject string
sl	sleep flag
sp	speed flag
sv	which spell is being cast
sw	swing flag
sz	silence flag



3	regeneration
4	invisibility
5	poisoning
6	sleeping
7	speed
8	healing
9	teleportation
>9	??? not used

6 = <item type dependent>

itemtype	meaning
<1	none
1	number of sides/die
2	none
3	none
>3	duration of magical power

7 = <item type dependent>

itemtype	meaning
<1	none
1	attacking bonus
2	defending bonus
3	none
>3	magic charges left

8 = light provision

value	meaning
-1	light forever
0	cannot provide
>1	burns for x number of turns

9 = item's whereabouts

value	meaning
<0	carried by -x monster
0	carried by player
>0	in room x

c\$(commands)

legal commands

- 1 = north
- 2 = east
- 3 = south
- 4 = west
- 5 = northeast
- 6 = southeast
- 7 = southwest
- 8 = northwest
- 9 = up
- 10 = down
- 11 = ne
- 12 = se
- 13 = sw



7 = weapon readied

value	meaning
<0	natural weapon where bonus = absolute value of value
0	no weapon
>0	using the item corresponding to the value

8 = reaction

value	meaning
<0	enemy of player
0	neutral
1	friendly
>2	joins player's party

9 = status

value	meaning
-2	berserk
-1	fleeing
0	normal
1	asleep
2	tied up
3	dead

Note that for values less than 1 the monster is active and for values greater than 0 the monster is incapacitated

10 = room

ranges from 1 to nr

11 = dice of damage if the monster has a natural attack

any number > 0

12 = sides per die

any number > 0

13 = introduction

value	meaning
0	never met
1	just met last turn
2	just met and described
3	reaction determined but not described
4	reaction determined and described

mn\$(monster)                      name of each monster

ranges from 1 to nm

rn\$(room)                          name of each room

ranges from 1 to nr

rd%(room, exit) exits for each room

(only exit subscript shown)

0 = room introduction

value	meaning
0	player has never been here before
-1	player has been here

1 = north

value	meaning
<0	secret exit leading to room = absolute value of value
0	no exit
1 - nr	leads to that room
999	exit out of adventure

2 = east  
 3 = south  
 4 = west  
 5 = northeast  
 6 = southeast  
 7 = southwest  
 8 = northwest  
 9 = up  
 10 = down

s\$(status+2) monster's status types

0 = berserk  
 1 = fleeing  
 2 = normal  
 3 = asleep  
 4 = tied up  
 5 = dead

sa%(spell) player's spell abilities

ranges from 0 - 100%

1 = speed spell  
 2 = heal spell  
 3 = blast spell  
 4 = power spell

## Formulas

Attacker's chance to hit in battle:

basic chance = attacker's agility \* 3 + attacker's weapon bonus  
- defender's armour bonus - defender's agility

Modifications:

IF no light THEN chance = chance / 2

IF attacker is berserk THEN add random 0-19 to chance

IF defender = player THEN any protection is subtracted; curse is added

IF attacker = player THEN chance = chance + 5 \* player's level

Once the final chance is computed, a percentage roll is made and:

IF roll > chance THEN attacker missed

IF roll <= chance THEN attacker hit

Note that a roll < 5 = a direct hit (double damage) while a roll > 98 = a fumble

### Fumble Recovery

100% chance if natural weapon

50% chance if man-made weapon

### Damage Taken

basic damage = (random 1 - sides/die), rolled dice of damage times and summed

Modifications:

damage = damage + 1/10 attacker weapon bonus - 1/10 defender armour bonus

damage = damage + attacker's strength

IF direct hit THEN damage = damage \* 2

Note that the weapon, if man-made, takes damage = 1/10 weapon bonus

Also that the armour, if man-made, takes damage = 1/10 armour bonus

## Health

health percentage = damage taken / max hit points

States:

health%      0-19 = ok  
                  20-39 = scratched  
                  40-59 = lightly wounded  
                  60-79 = wounded  
                  80-89 = heavily wounded  
                  90-99 = critical  
                  100 = killed

## Experience & Level

player experience = experience + 2 \* defeated enemy's max hit points

player level = experience / 500

## Friendliness & Reaction Determination

Monster's friendliness = initial percentage (an attribute)

Modifications:

Friendliness = friendliness + player's charisma (if current room has light)

Comparison with two random percentage rolls and conversion to -1 to 2 scale:

IF friendliness > roll #1 & > roll #2 THEN friendliness = 2  
 IF friendliness > roll #1 & ≤ roll #2 THEN friendliness = 1  
 IF friendliness ≤ roll #1 & > roll #2 THEN friendliness = 0  
 IF friendliness ≤ roll #1 & ≤ roll #2 THEN friendliness = -1

Monster's reaction = initial value (an attribute) where:

-32768 - -1 = mad (will attack on sight)  
 0 = neutral (will not attack player)  
 1 = friendly (will fight with player in current room but not leave)  
 2 - 32767 = joins party (will fight with player in current room and leave with him)

Modifications:

Monster's reaction = monster's reaction + friendliness

Can also be modified by player actions (gifts of money and/or items)



## Appendix C: Adventure Program Reference

This appendix contains a wealth of information about how an *Imagery!* adventure actually works. Study this appendix to gain an understanding of the inner workings of your adventures, especially if you want to make special modifications.

### Program Overview

The *Imagery!* base adventure program ("adventure") is made up of 6 major parts:

- 1) initialization routine
- 2) main loop (command processor)
- 3) combat routines
- 4) magic routines
- 5) routines for each command
- 6) assorted utility routines

Execution is passed from either the Towne program ("towne") on the *Imagery!* boot disk or from the program "title" on the adventure disk. The first few lines (65-80) determine whether a new or saved game is being played. If a saved game is being restarted, the variables are read back in and execution jumps to the Main Loop (130).

If a new game is being started, execution jumps to the initialization routine at 3500 and then to the Main Loop.

Most of the execution time is spent in the Main Loop processing commands from the player. Branches are made to carry out the player's commands.

### Data Files

Each base adventure program has eight unique data files, each of which contains a different part of the information making up the adventure. The files are:

- 1) advent.name - adventure name, author's name, and intro text. Note that this file is not used by "adventure"—only by the "towne" and "title" programs.
- 2) advent.desc - number of rooms, monsters, & items in the adventure
- 3) room.name - short description, light status, and exits for each room
- 4) room.desc - long room descriptions
- 5) monster.name - monster names & attributes
- 6) monster.desc - monster descriptions
- 7) item.name - item names, effects, & attributes
- 8) item.desc - item descriptions

The player's data is stored in a temporary file with the name of the character preceded

by a "+". It is erased as soon as the data is read in.

## New Commands

New commands can be added to "adventure" after line 7700. Please make sure you have a very good understanding of how everything works before trying this. Notice that you will need to add any new commands to the command list as well as increase the number of legal commands (lines 3700-3740).

Keep in mind that existing commands often can be modified rather than creating new ones. Try to avoid duplicate commands or ones that are used only once or twice in the adventure.

## Pseudo-Code Listing

```

65      PRINT color black:CLR variables:OPEN error channel:PEEK character
        number:OPEN save game file (V=0 for new game, file will never be
        found)
70      INPUT from error channel:IF file not found THEN CLOSE file:GOTO
        3500 (start new game)
75      IF any other error then CLOSE file:CLOSE error channel:END (This is
        not supposed to happen!)
80      SYS ML routine to read variables back in:CLOSE file:scratch save game
        file:GOTO 130 (resume saved game)

100     *** MAIN LOOP ***

110     PRINT press space
120     GET keypress:IF no press THEN 120
130     PRINT clear screen
140     GOSUB 1000 (magic routines)
150     GOSUB 1300 (light routine)
160     GOSUB 1400 (room description routine)
170     GOSUB 1500 (move monsters routine)
180     GOSUB 1600 (display monsters routine)
190     GOSUB 1900 (display artifacts routine)
200     GOSUB 2000 (check combat routine)

500     *** INPUT COMMAND FROM PLAYER ***

510     IF player is asleep THEN PRINT sleep:GOTO 110
520     INPUT command string:IF this is the same command as last time THEN
        PRINT command (This is if player just presses RETURN)
530     Set last command = current command
540     IF the leftmost character is a space THEN remove it:GOTO 540
550     Initialize counter
560     IF the space separating the words is found THEN 590
570     Increment counter:IF not at end of command string THEN 560

```

```
580     Verb string = left portion of command string:clear counter:subject string =
      null:GOTO 600
590     Verb string = left portion of command string:subject string = right portion
      of command string:clear counter
600     IF leftmost character of subject string is a space THEN remove it:GOTO
      600
610     Increment counter:IF left portion of a legal command = command string
      THEN 640
620     IF counter < number of legal commands THEN 610
630     PRINT can't understand:GOTO 520
640     IF the command number < 15 THEN 5000 (This is a movement command)
650     Subtract 14 from command number:ON command number GOTO
      5100,5200,5300,5400,5600,5800,5900,6000,6100,6300,6400,6500
660     Subtract 12 from command number:ON command number GOTO
      6600,6700,6800,7100,7200,7300,7400,7500,7600
670     STOP (This line should never be reached)

1000    *** MAGIC/POWERS ROUTINE ***

1010    IF no speed THEN 1050
1020    Decrement speed counter:IF still speed left THEN 1050
1030    Player's agility is now halved (back to normal)
1040    IF random chance THEN PRINT speed expired
1050    IF no protection THEN 1080
1060    Decrement protection:IF still protected THEN 1080
1070    IF random chance THEN PRINT protection wore off
1080    IF no curse THEN 1110
1090    Decrement curse:IF still cursed THEN 1110
1100    IF random chance THEN PRINT curse gone
1110    IF not invisible THEN 1140
1120    Decrement invisibility:IF still invisible THEN 1140
1130    PRINT now visible
1140    IF no poison THEN 1180
1150    Decrement poison:increment player's damage
1155    IF player's damage = player's health THEN PRINT poison kills
      you:GOTO 3070 (game over)
1160    IF still poisoned THEN 1180
1170    IF random chance THEN PRINT poison gone
1180    IF not regenerating THEN 1200
1190    Decrement regeneration:decrement player's damage:
1195    IF no damage left THEN stop regenerating:damage = 0:IF random chance
      THEN PRINT regenerated
1200    IF not asleep THEN RETURN
1210    Decrement sleep:IF still asleep THEN RETURN
1220    PRINT wake up:RETURN

1300    *** LIGHT ROUTINE ***

1310    Light flag = dark:IF room is naturally lit THEN light flag = light
1315    FOR item = 1 TO number of items in adventure:IF item is not lit THEN
      1350
1320    IF item is lit (but not forever) THEN decrement burn counter:IF item
      burned out THEN 1360
```

```
1340 IF item is in this room OR player is carrying it THEN light flag = light
1350 NEXT item:RETURN
1360 IF item is in room OR player is carrying it THEN PRINT it went out
1370 Item's burn counter = 0:item's light = out:GOTO 1350

1400 *** ROOM DESCRIPTION ***

1410 IF no light THEN PRINT too dark:RETURN
1420 IF player has been here before THEN PRINT short description:GOTO
1440
1430 OPEN room description file:record pointer = room:GOSUB 4800:set flag
that player has been here before
1440 PRINT exits:FOR exit = 1 TO 10
1450 IF visible exit in this direction THEN PRINT exit
1460 NEXT exit:PRINT:RETURN

1500 *** MOVE MONSTERS ***

1510 FOR monster = 1 TO total number of monsters:IF monster's reaction =
friendly THEN IF monster's status = normal THEN monster's room =
current room
1530 IF monster's status = berserk AND monster's room <> current room
THEN 1580
1540 IF monster's room <> current room OR 10% chance OR monster's status
<> fleeing THEN 1570
1550 Direction = random number from 0 to 11:IF direction = 11 THEN 1570
(don't move at all)
1555 IF no exit in that direction THEN 1550 (try another direction)
1560 PRINT monster flees:monster's room = room out the exit
1570 NEXT monster:RETURN
1580 Monster's room = current room:IF monster's reaction is bad THEN
PRINT monster chasing:GOTO 1570
1590 Monster's reaction = normal:GOTO 1570

1600 *** DISPLAY MONSTERS ***

1610 Battle flag = off:FOR monster = 1 TO total number of monsters:IF
monster's room <> current room THEN 1700
1620 IF monster's reaction = bad AND monster's status = normal or berserk
AND player is not invisible THEN battle flag = on
1630 Temp = monster's introduction:description flag = off:IF temp = just met
OR temp = described only AND monster's status = aware AND player is
not invisible THEN GOSUB 1750:temp = temp + 2
1640 IF temp = never met THEN temp = just met
1650 IF temp = just met OR temp = reaction determined only THEN IF light
flag = light THEN increment temp:GOTO 1710
1670 IF light THEN PRINT monster's status; monster is here
1680 IF reaction was determined THEN GOSUB 1830
1690 Monster's introduction = temp
1700 NEXT monster:PRINT:RETURN
1710 OPEN monster description file:record pointer = monster:GOSUB 4800
(print monster desc):GOTO 1670
```

```
1750      *** DETERMINE MONSTER'S REACTIONS ***
1760      Friendliness = monster's friendliness plus player's charisma (if light):IF
friendliness > random chance THEN 1790
1770      IF friendliness > random chance THEN friendliness = neutral:GOTO 1810
1780      Friendliness = mean:GOTO 1810
1790      IF friendliness > random chance THEN friendliness = joins party:GOTO
1810
1800      Friendliness = friends
1810      Monster's reaction = old reaction + friendliness:IF light THEN desc flag =
on
1820      RETURN
1830      IF monster's reaction < 0 THEN PRINT monster mad:RETURN
1840      IF monster's reaction = 0 THEN PRINT monster neutral:RETURN
1850      IF monster's reaction = 1 THEN PRINT monster friends:RETURN
1860      IF monster's reaction > 1 THEN PRINT monster joins party:RETURN

1900      *** ITEM DISPLAY ***

1910      IF no light THEN RETURN
1920      FOR item = 1 TO total number of items:IF item's room = current room
THEN 1940
1930      NEXT item:RETURN
1940      PRINT broken or nothing
1950      PRINT lit or nothing; item is here:GOTO 1930

2000      *** COMBAT ROUTINE ***

2010      IF no battle THEN RETURN
2020      Battle = off:number of friends in battle = 0:number of enemies in battle =
0:FOR monster = 1 TO total number of monsters
2030      IF monster's reaction = friendly AND monster's room = current room
AND monster's status = aware THEN increment number of friends:store
in friend array
2040      IF monster's reaction = bad AND monster's room = current room AND
monster's status = aware THEN increment number of enemies:store in
enemy array
2050      NEXT monster:IF no enemies THEN 2070
2060      FOR Z = 1 TO number of enemies:attacker = enemy array:defender = ran-
dom friend:GOSUB 2100:NEXT Z
2070      IF no friends THEN 2090
2080      FOR Z = 1 TO number of friends:attacker = friend array:defender = ran-
dom enemy:GOSUB 2100:NEXT Z
2090      RETURN

2100      *** ATTACK ROUTINE ***

2110      GOSUB 2200 (check if it's legal to swing)
2120      IF swing allowed THEN GOSUB 2400 (swing weapon)
2130      IF hit flag < 0 THEN GOSUB 2900 (fumble)
2140      IF hit flag > 0 THEN GOSUB 2600 (apply damage)
2150      IF swing allowed THEN GOSUB 2800 (damage report)
```

```

2160 IF killed THEN GOSUB 3000 (death occurred)
2170 IF flee THEN GOSUB 3200 (fleeing monster)
2180 RETURN

2200 *** WEAPON & CHECK ***

2210 Swing flag = off:hit flag = off:temp = 0:IF attacker's status = fleeing OR
status = sleep, tied, dead OR defender's status = dead THEN RETURN
2230 IF monster's weapon = ready THEN swing flag = on:weapon = monster's
weapon:RETURN
2240 PRINT no weapon:IF attacker = player THEN RETURN
2250 Increment temp:IF item's room = current room OR item's room = attacker
(attacker is carrying it) THEN 2300
2260 IF temp < total number of items THEN 2250
2270 IF high random chance OR the defender does not have a weapon THEN
PRINT can't find a weapon!RETURN
2280 Attacker's weapon = defender's weapon:defender's weapon =
nothing:weapon's room = attacker
2285 IF the defender is the player THEN subtract the weight of the weapon
2290 PRINT weapon stolen!RETURN
2300 IF item is broken OR item's type <> weapon THEN 2260
2310 PRINT attacker readies weapon:weapon's room = attacker:attacker's
weapon = weapon:RETURN

2400 *** SWING ROUTINE ***

2410 PRINT attacker attacks defender:weapon bonus = attacker's weapon
2420 IF natural weapon THEN make bonus positive:PRINT how monster
attacks:GOTO 2440
2430 Bonus = weapon's bonus:PRINT with weapon name:PRINT weapon's
effect
2440 Armour bonus = monster's armour:IF armour is natural THEN make
bonus positive:GOTO 2460
2450 Armour bonus = defender's armour's bonus
2460 Chance to hit = 3 times attacker's agility + weapon bonus - armour bonus -
defender's agility:IF no light THEN chance is halved
2470 If defender = player and player is protected then subtract protection from
chance
2490 If attacker = player and player is cursed then add curse:if attacker is ber-
serk then add random 0-19
2510 IF attacker = player THEN chance = chance + 5 times player's level
2520 Roll = random number from 0-100:IF roll > 98 THEN hit = -1:PRINT
fumble:RETURN
2530 IF roll > chance THEN PRINT miss:hit = 0:RETURN
2540 IF roll < 5 THEN PRINT direct hit:hit = 2:RETURN
2550 PRINT hit:hit = 1:RETURN

2600 *** DAMAGE ROUTINE ***

2610 Damage = 0:IF weapon is natural THEN dice = monster's dice:sides =
monster's sides:GOTO 2630
2620 Dice = weapon's dice:sides = weapon's sides

```

```
2630   FOR temp = 1 TO dice: damage = random roll: NEXT temp: damage = dam-
      age + tenth of weapon's bonus - tenth of armour's bonus
2635   Damage = damage + attacker's strength bonus: damage = damage times hit
      (1=normal or 2=double): IF defender's status = sleeping THEN status =
      normal
2640   IF no damage THEN PRINT blow bounces: GOTO 2655
2650   Monster's damage = monster's damage + damage
2655   IF weapon is natural THEN 2670
2660   Item's points = item's points - tenth of bonus
2665   IF no more points THEN PRINT it broke!: attacker's weapon = 0: item's
      light = off
2670   Armour = defender's armour: IF armour is natural THEN RETURN
2680   Item's points = item's points - tenth of bonus
2690   IF no more points THEN PRINT armour falls off!: monster's armour = 0
2700   RETURN

2800   *** DAMAGE REPORT ***

2810   Health percentage = monster's damage / monster's health
2820   Kill flag = off: flee flag = off: IF monster's courage < health% AND
      defender <> player THEN flee flag = on
2830   IF health% > 99 THEN PRINT killed: kill flag = on: RETURN
2840   IF health% > 89 THEN PRINT critical: RETURN
2850   IF health% > 79 THEN PRINT heavily wounded: RETURN
2860   IF health% > 59 THEN PRINT wounded: RETURN
2870   IF health% > 39 THEN PRINT lightly wounded: RETURN
2880   IF health% > 19 THEN PRINT scratched: RETURN
2890   PRINT good health: RETURN

2900   *** FUMBLE ROUTINE ***

2910   IF random 50% chance OR monster's weapon is natural THEN PRINT
      recovered: RETURN
2930   PRINT dropped!: weapon's room = current room: monster's weapon =
      none: IF attacker = player THEN weight carried = weight - weapon's
      weight
2940   Weapon's light = off: RETURN

3000   *** DEFENDER KILLED ***

3010   IF attacker = player THEN GOSUB 3300 (experience)
3020   Monster's status = dead meat: FOR temp = 1 TO total number of items: IF
      item's room = defender THEN item's room = current room
3030   NEXT temp: flee flag = off: kill flag = off: IF defender <> player THEN
      RETURN
3040   PRINT press space
3050   GET keypress: IF no press THEN 3050
3060   PRINT you're dead!
3070   PRINT insert Imagery! disk
3080   PRINT press F1
3090   GET keypress: IF key <> F1 THEN 3090
3100   GOSUB 4900: CLOSE error channel: PRINT load title
```

```

3110     POKE returns into keyboard buffer:END

3200     *** FLEE ROUTINE ***

3210     Roll = random:IF roll < 20% THEN 3270
3220     IF roll < 40% THEN PRINT freezes!:monster's status = normal:RETURN
3230     IF roll < 65% THEN RETURN (no effect)
3240     Direction = random 0-11:IF direction = 11 THEN RETURN
3250     PRINT defender flees
3260     Defender's room = room out exit:monster's status = fleeing:RETURN
3270     PRINT defender goes berserk!:monster's courage = 100:monster's status =
        berserk
3280     RETURN

3300     *** GAIN EXPERIENCE ***

3310     Exp = exp + twice defender's max health:PRINT points earned
3320     Temp = current level:level = exp / 500:IF new level = old level THEN
        RETURN
3330     PRINT raised a level!:temp = random 2-8
3340     PRINT hit points gained:player's hp = player's hp + temp
3350     Flash border:RETURN

3500     *** INITIALIZE GAME ***

3510     PRINT waking monsters
3520     OPEN advent.desc file:INPUT number of items, number of monsters,
        number of rooms:CLOSE file
3530     DIM array for room names, room exits, room light, monster attributes,
        monster names, monster attack descriptions
3540     OPEN characters file:INPUT player's name, experience, gold, gold in bank,
        spell 1
3550     INPUT spell 2, spell 3, spell 4, player's strength, agility, charisma, hit points
3560     INPUT number of items carried:total number of items = items carried +
        items in adventure:DIM array for item names, item attributes, item
        effects:IF total number of items = number of items from the adventure
        (player has no items) THEN 3580
3570     FOR item = items in adventure + 1 TO total items:INPUT item's name,
        item's effect:FOR temp = 1 TO 9:INPUT item's attributes:NEXT temp,
        item
3580     CLOSE file:PRINT scratch file
3590     OPEN monster file:FOR temp = 1 TO total monsters:INPUT monster's
        name, monster's attack desc
3600     FOR temp = 1 TO 12:INPUT monster's attributes:NEXT temp,
        monster:CLOSE file
3610     OPEN room file:FOR room = 1 TO total rooms:INPUT room's name,
        room's light
3620     FOR temp = 1 TO 10:INPUT room's exits:NEXT temp, room:CLOSE file
3630     OPEN item file:FOR item = 1 TO total items (not carried):INPUT item's
        name, item's effect
3640     FOR temp = 1 TO 9:INPUT item's attributes:NEXT temp, item:CLOSE file
3650     Current room = 1:non-existent items = nothing:player's level = exp / 500

```



```
3660      Status codes (add 2 to each array since subscripts can't be negative) -2 =
          berserking, -1 = fleeing, (0 = normal), 1 = sleeping
3670      2 = tied up, 3 = dead
3680      Weight carried = 0:IF no items carried THEN 3700
3690      FOR item = items not carried + 1 TO total items:weight = weight + item's
          weight:NEXT item:weight = weight + amount of gold / 100
3700      READ number of commands:DIM command array:FOR temp = 1 TO total
          commands:READ command:NEXT temp:GOSUB 4700:GOTO 130
3710      DATA number of commands
3720      DATA directions (these MUST be the first 14 data items for the move rou-
          tine to work)
3730      DATA commands
3740      DATA commands

4000      *** READ EFFECTS ***

4010      OPEN effects file:record pointer = effect:GOSUB 4800 (print item
          effect):RETURN

4100      *** ITEM NUMBER ***

4110      Item = max item:IF subject = null THEN PRINT verb:INPUT
          subject:GOTO 4110
4120      IF leftmost character = $ (for gold) THEN item = -:RETURN
4130      IF left portion of item's name = subject THEN RETURN
4140      Decrement item:IF more to check THEN 4130
4150      PRINT not recognized:RETURN

4200      *** CHECK FOR ALLOW ***

4210      Allow flag = on:monster = 0:IF invisible flag = on THEN RETURN
4220      Increment monster:IF monster's room = current room AND monster's
          status = aware THEN 4250
4230      IF more to check THEN 4220
4240      RETURN
4250      IF monster's intro shows reaction not determined yet OR monster's reac-
          tion is bad THEN allow = off:RETURN
4260      GOTO 4230

4300      *** MONSTER NUMBER ***

4310      Monster = -:IF subject = null THEN PRINT verb:INPUT subject:GOTO
          4310
4320      Increment monster:IF left portion of monster's name = subject THEN
          RETURN
4330      IF more to check THEN 4320
4340      Monster = -:PRINT never heard:RETURN

4400      *** CAST SPELL ***

4410      Temp = off:IF random chance < spell ability THEN temp = on:RETURN
4420      PRINT failed:RETURN
```

```
4430      *** REDUCE SPELL CHANCE ***
4440      Spell ability = spell ability / 2:GOTO 110
4500      *** LEAVING ADVENTURE ***
4510      PRINT you want to?
4520      GET keypress:IF no press THEN 4520
4530      IF keypress <> y THEN current room = last room:GOTO 130
4540      PRINT made it out!!
4550      PRINT insert disk
4560      PRINT press F1
4570      GET keypress:IF press <> F1 THEN 4570
4580      Items carried = 0:FOR item = 1 TO total items:IF item's type = treasure
      and item's room = player THEN gold = gold + treasure value:item's room
      = first room (drop it)
4590      IF item's room = player THEN increment items carried
4600      NEXT item:GOSUB 4900:OPEN character file:PRINT player name
4610      PRINT exp, gold, gold in bank:FOR spell = 1 TO 4:IF some spell ability
      left THEN ability = 100% (full strength)
4620      PRINT spell ability:NEXT spell:FOR temp = 1 TO 3:PRINT player's
      attributes:NEXT temp:PRINT hit points
4630      PRINT items carried:IF none THEN CLOSE file:GOTO 4670
4640      FOR item = 1 TO total items:IF item is not owned THEN 4660
4650      PRINT item's name, item's effect:FOR temp = 1 TO 9:PRINT item's
      attributes:NEXT temp
4660      NEXT item:CLOSE file:CLOSE error channel
4670      PRINT load town
4680      POKE returns into keyboard buffer
4700      *** STRIP # SIGNS ***
4710      FOR item = 1 TO total items
4720      IF leftmost character in name = # THEN remove #:GOTO 4720
4730      NEXT item
4740      *** ADD # SIGNS ***
4750      FOR item = total TO second STEP -1:count = 0:FOR temp = item TO first
      item STEP -1
4760      IF item's name = another item's name THEN increment count:add count
      number of # signs to name
4770      NEXT temp, item:RETURN
4800      *** POSITION POINTER ***
4810      PRINT pointer:SYS ML to read record:CLOSE file:RETURN
4900      *** CHECK DISK ***
4910      OPEN buffer:PRINT read sector:PRINT move pointer
4920      GET disk ID:CLOSE buffer:IF ID = IM THEN RETURN
```

```
4930 PRINT insert disk
4940 Pause:PRINT erase
4950 GET key:IF no key THEN 4930
4960 GOTO 4910

5000 *** MOVE ***

5010 IF no light THEN PRINT scary in dark
5020 IF command > 10 THEN command = command - 6 (for abbreviations)
5030 Temp = room's exit:IF temp is not a legal exit THEN PRINT can't go that
way:GOTO 110
5040 IF old room <> temp THEN GOSUB 4200:IF NOT allowed THEN PRINT
monster blocks exit:GOTO 110
5050 Old room = current room:current room = temp:IF current room = 999
THEN PRINT congrats!:GOTO 4500 (made it out)
5060 GOTO 130

5100 *** LOOK ***

5110 IF NOT light THEN PRINT it's black:GOTO 110
5120 Room's intro = 0 (never been in room):GOTO 130

5200 *** EXAMINE ITEM ***

5210 IF NOT light THEN PRINT hard to see it:GOTO 110
5220 GOSUB 4100 (get item number):IF no item THEN 110
5225 IF item = -1 THEN PRINT it's money:GOTO 110
5230 IF item's room is not player AND item's room <> current room THEN
PRINT can't find it:GOTO 110
5240 IF item was not brought into the adventure by the player THEN OPEN
item desc file:PRINT:record pointer = item:GOSUB 4800 (print item
desc):GOTO 110
5250 PRINT can only examine items found in adventure:GOTO 110

5300 *** PLAYER ATTACK ***

5310 IF NOT light THEN PRINT swing in the dark:GOTO 110
5320 GOSUB 4300 (check for legal target):IF not a legal monster to attack
THEN 110
5330 IF monster's room <> current room THEN PRINT suffering from
delusions?:GOTO 110
5340 IF monster's status = dead THEN PRINT it's dead all ready!:GOTO 110
5350 Attacker = player:Defender = monster:monster's reaction = monster's reac-
tion - 5:GOSUB 2100 (attack):GOTO 110

5400 *** GET ITEM ***

5410 IF NOT light THEN PRINT bumped your head:GOTO 110
5420 GOSUB 4100 (get item number):IF no item THEN 110
5430 IF item < 0 THEN PRINT no money here:GOTO 110
5440 IF item's room <> current room THEN PRINT not here:GOTO 110
5450 IF item's weight = 9999 THEN PRINT leave it alone:GOTO 110
```

```
5460 GOSUB 4200 (check for allow):IF allowed THEN 5480
5470 IF item's type isn't normal THEN PRINT monster won't let you:GOTO
110
5480 IF weight + item's weight > 8 times player's strength THEN PRINT too
heavy:GOTO 110
5490 Item's room = player:PRINT got it:weight = weight + item's weight
5500 IF player doesn't have a weapon ready AND item's type = weapon THEN
player's weapon = item:PRINT item is ready
5510 GOTO 110

5600 *** DROP ITEM ***

5610 GOSUB 4100 (get item number):IF no item THEN 110
5620 IF item = -1 THEN amount = value of subject:GOTO 5690 (this is money)
5630 IF item isn't carried THEN PRINT you're not carrying it:GOTO 110
5640 Item's room = current room:weight = weight - item's weight:PRINT
dropped
5650 IF NOT light THEN PRINT dropped on your toe:item's hit points =
item's hit points - 5
5660 IF item was player's weapon THEN player's weapon = nothing:item's light
= 0 (if a magical weapon, extinguish it)
5670 IF item was player's armour THEN player's armour = nothing
5680 GOTO 110
5690 IF amount > gold THEN PRINT don't have that much:GOTO 110
5700 Gold = gold - amount:weight = weight - amount / 100:PRINT it rolls
away:GOTO 110

5800 *** LIGHT ITEM ***

5810 GOSUB 4100 (get item number):IF no item THEN 110
5815 IF item = money THEN PRINT it burns up:weight = weight - gold /
100:gold = 0:GOTO 110
5820 IF item isn't carried THEN PRINT you don't have it:GOTO 110
5830 IF random 10% chance THEN PRINT you burned your finger:GOTO 110
5840 IF item's hit points < 0 THEN PRINT it's broken:GOTO 110
5850 IF item will provide light THEN item's light = lit:PRINT item's
effect:GOTO 110
5860 PRINT it doesn't burn:GOTO 110

5900 *** READY WEAPON ***

5910 GOSUB 4100 (get item number):IF no item THEN 110
5915 IF item = -1 THEN PRINT give monsters money instead:GOTO 110
5920 IF item isn't carried THEN PRINT don't have it:GOTO 110
5930 IF item's type <> weapon THEN PRINT it isn't a weapon:GOTO 110
5940 IF item's hit points < 1 THEN PRINT it's broken:GOTO 110
5950 IF weapon's magic = 0 THEN 5970 (doesn't glow magically)
5960 PRINT weapon glows:item's light = lit
5970 Readied weapon's room = player:weapon ready = weapon:PRINT weapon
ready:GOTO 110

6000 *** WEAR ARMOUR ***
```

```
6010 GOSUB 4100 (get item number):IF no item THEN 110
6015 IF item = -1 THEN PRINT that's a new one (can't wear money):GOTO 110
6020 IF item isn't carried THEN PRINT don't have it:GOTO 110
6030 IF item's type <> armour THEN PRINT that's not armour:GOTO 110
6040 IF item's hit points < 1 THEN PRINT it's broken:GOTO 110
6050 Armour worn = armour:PRINT wearing armour:GOTO 110

6100 *** USE ITEM ***

6110 GOSUB 4100 (get item number):IF no item THEN 110
6115 IF item = -1 THEN PRINT money isn't magic:GOTO 110
6120 IF item's hit points < 1 THEN PRINT can't use a broken item:GOTO 110
6130 IF item's room <> current room AND item isn't carried THEN PRINT
can't find it:GOTO 110
6140 GOSUB 4200 (check for allow):IF allowed OR if item is carried THEN
6160
6150 PRINT monster won't let you use it:GOTO 110
6160 IF item's type isn't magical THEN PRINT nothing happens:GOTO 110
6170 IF item's charge = 0 THEN PRINT it fizzles:GOTO 110
6180 Decrement item's charge:PRINT description of item's effect
6190 ON item's effect GOTO 6200,6210,6220,6230,6240,6250,6260,6280,6290
6200 Protection = item's power:PRINT protected:GOTO 110
6210 Curse = item's power:PRINT cursed:GOTO 110
6220 Regeneration = item's power:PRINT regenerating:GOTO 110
6230 Invisibility = item's power:PRINT invisible:GOTO 110
6240 Poison = item's power:PRINT poisoned:GOTO 110
6250 Sleep = item's power:PRINT fall asleep:GOTO 110
6260 IF speed = off THEN speed = item's power:PRINT faster:player's dex =
player's dex times 2
6270 PRINT can't make you go faster:GOTO 110
6280 Player's damage = 0:PRINT healed:GOTO 110
6290 Room = random room in adventure:PRINT teleported:GOTO 110

6300 *** SPEED SPELL ***

6310 Spell = 1:GOSUB 4400 (cast spell):IF spell failed THEN 110
6320 IF already fast THEN PRINT already fast:GOTO 110
6330 Speed = random 0-25:dex = dex times 2:PRINT fast:GOTO 4430

6400 *** HEAL SPELL ***

6410 Spell = 2:GOSUB 4400 (cast spell):IF spell failed THEN 110
6420 GOSUB 4300 (check for legal target):IF illegal monster THEN 110
6430 IF monster's room <> current room THEN PRINT monster isn't
here:GOTO 110
6440 IF monster's status = dead THEN PRINT only priest can resurrect:GOTO
110
6450 Monster's damage = monster's damage - random 0-20:IF monster's damage
< 0 THEN monster's damage = 0
6460 IF monster's status = fleeing THEN monster's status = normal
6470 PRINT monster is healed:defender = monster:GOSUB 2800 (damage
report):GOTO 4430
```

```

6500    *** BLAST SPELL ***

6510    IF NOT light THEN PRINT it's too dark:GOTO 110
6520    Spell = 3:GOSUB 4400 (cast spell):IF spell failed THEN 110
6530    GOSUB 4300 (check for legal target):IF illegal monster THEN 110
6540    IF monster's room <> current room THEN PRINT it isn't here:GOTO 110
6550    PRINT you blast him
6560    Monster's reaction = monster's reaction - 5:monster's damage = monster's
        damage + 0-10 times player's level
6570    Defender = monster:GOSUB 2150 (damage report):GOTO 4430

6600    *** POWER SPELL ***

6610    Spell = 4:GOSUB 4400 (cast spell):IF spell failed THEN 110
6620    Roll = random:IF roll < 10% THEN PRINT cursed:curse = 25 (number of
        turns it lasts):GOTO 4430
6630    IF roll < 20% THEN PRINT protected:protect = 25:GOTO 4430
6640    IF roll < 30% THEN PRINT regenerating:regenerate = 25:GOTO 4430
6650    IF roll < 40% THEN PRINT poisoned:poison = 25:GOTO 4430
6660    IF roll < 50% THEN PRINT invisible:invisible = 10:GOTO 4430
6670    IF roll < 60% THEN PRINT asleep:sleep = 10:GOTO 4430
6680    PRINT no effect (40% chance)
6690    GOTO 4430

6700    *** INVENTORY ***

6710    PRINT carrying
6720    FOR item = 1 TO total items:IF item's room = player THEN 6750
6730    NEXT item:PRINT weapon; readied weapon name
6740    PRINT wearing; armour name:PRINT money; gold:GOTO 520 (doesn't
        take a turn)
6750    PRINT whether item is broken or not
6760    PRINT whether item is lit or not; item's name:GOTO 6730

6800    *** GIVE ***

6810    GOSUB 4100 (get item number):IF no item THEN 110
6820    IF item = -1 THEN 7010 (money)
6830    IF item not carried THEN PRINT don't have it:GOTO 110
6840    INPUT to who:GOSUB 4300:IF illegal monster THEN PRINT feeling
        ok?:GOTO 110
6850    Temp = 0:IF monster's room <> current room THEN PRINT monster
        isn't here:GOTO 110
6860    IF monster's status is disabled THEN PRINT monster is
        incapacitated:GOTO 110
6870    PRINT monster says:silence = on
6875    IF item = -1 (money) THEN PRINT thanks:GOTO 7030
6880    IF item's hit points < 1 THEN PRINT broken:temp = -1:silence = off
6890    IF item's weight > 50 THEN PRINT hate heavy stuff:decrement
        temp:silence = off
6900    IF item's type = normal THEN PRINT junk:decrement temp:silence = off
6910    IF item's value > 100 THEN PRINT valuable:increment temp:silence = off

```

```
6920     IF item's weight < 1 THEN PRINT light:increment temp:silence = off
6930     IF item's type is treasure or magic THEN PRINT nice:silence = off
6940     IF nothing said THEN PRINT nothing
6950     Monster's reaction = monster's reaction + temp:IF temp is non-zero
        THEN GOSUB 1830 (determine new reaction)
6960     IF item was player's weapon THEN player has no weapon readied:weapon
        is no longer lit
6970     IF item was player's armour THEN player has no armour
6980     IF monster's reaction is bad THEN 7000
6990     Item's room = monster:weight = weight - item's weight:GOTO 110
7000     Item's room = current room:weight = weight - item's weight
7005     PRINT thrown back in face:GOTO 110
7010     Temp = value of subject:IF temp > gold THEN PRINT don't have that
        much:GOTO 110
7020     GOTO 6840
7030     Monster's reaction = monster's reaction + amount offered / 250:IF amount
        / 250 is positive THEN GOSUB 1830 (determine new reaction)
7040     Gold = gold - amount:weight = weight - amount / 100:GOTO 110

7100     *** SEARCH ROOM ***

7110     IF NOT light THEN PRINT you trip:GOTO 110
7120     GOSUB 4200 (check for allow):IF NOT allowed THEN PRINT monster
        steps in the way:GOTO 110
7130     Silence = off:FOR direction = 1 TO 10:IF exit is secret AND random
        chance THEN 7160
7140     NEXT direction:IF NOT silent THEN PRINT nothing found
7150     GOTO 110
7160     PRINT find secret passage:silence = off:exit = normal:GOTO 7140

7200     *** HELP ***

7210     PRINT here is list
7220     FOR command = 15 TO total commands + 1 STEP 2:PRINT command,
        following command:NEXT command:GOTO 520 (doesn't take a turn)

7300     *** STATUS ***

7310     PRINT:IF poisoned THEN PRINT poisoned
7320     IF cursed THEN PRINT cursed
7330     IF invisible THEN PRINT invisible
7340     IF protected THEN PRINT protected
7350     IF regenerating THEN PRINT regenerating
7360     IF speed THEN PRINT speed
7370     Defender = player:GOSUB 2800 (damage report):GOTO 520 (doesn't take a
        turn)

7400     *** SMILE ***

7410     PRINT:FOR monster = 1 TO total monsters:IF monster's room = current
        room THEN GOSUB 1830 (print current monster reactions)
7420     NEXT monster:GOTO 520 (doesn't take a turn)
```

```
7500      *** EXTINGUISH ***

7510      GOSUB 4100 (get item number):IF no item THEN 110
7515      IF item = -1 THEN PRINT too late (money burns instantly):GOTO 110
7520      IF item isn't carried THEN PRINT don't have it:GOTO 110
7530      IF item is lit THEN PRINT put it out:item's light = out:GOTO 110
7540      PRINT not lit:GOTO 110

7600      *** SAVE GAME ***

7610      PRINT save?
7606      GET keypress:IF not legal press THEN 7606
7607      PRINT press
7608      IF press is no THEN 130
7610      FOR temp = 0 TO 9:OPEN save file:INPUT error channel:IF no error
          THEN 7630 (valid file)
7620      CLOSE:NEXT temp:PRINT sorry:GOTO 110
7630      SYS ML to save variables:CLOSE file
7640      INPUT error channel:IF no error THEN PRINT game saved:CLOSE error
          channel:NEW
7650      PRINT error; bad save:GOTO 110

7700      *** ADD NEW COMMANDS HERE ***
```



## Appendix D: Flexi-Copy Operation

*Flexi-Copy* is an easy-to-use file copier for one or two disk drives. You can use it to copy files when you are working on an adventure. Many file copiers can't copy relative files, a file-type *Imagery!* uses, but *Flexi-Copy* can.

*Flexi-Copy* is on your designer disk. To use it, first switch your computer off then on to reset it. Then type:

```
LOAD "FLEXI-COPY",8 RETURN  
RUN RETURN
```

You will then see the opening screen. If you are going to use two disk drives, make sure the other one is turned on and set to the proper device number.

For each file you want to copy, you will be asked four questions:

- 1) Input device
- 2) Output device
- 3) Input filename
- 4) Output filename

The input device is the disk drive that has the source disk (the disk you want to copy from). The output device is the disk drive that has the target disk (the disk you want to copy to). The input device is usually 8, and the output 8 or 9, depending whether or not you are using two disk drives.

The input filename is the name of the file you want to copy. The output filename is the same as the input, so just press RETURN to keep the same name.

After answering these questions, *Flexi-Copy* will begin copying. If you are using one disk drive, the program will prompt you when it's ready for the target disk. If you are using two drives, the copying will proceed automatically.

That's all there is to it!

Note: When you're copying relative files, such as "room.desc", *Flexi-Copy* will display some extra information such as "Record length" and "Sectors". That's normal— it's useful stuff to know in certain situations.

## Appendix E: Submitting Imagery! Adventures

One of the best parts of the *Imagery!* system is that many authors can write adventures, not just one or two with the "inside" information. Once you have completed an adventure, you can give it to a friend without worrying about breaking any copyright laws.

*Imagination*, the official *Imagery!* club, is the central distribution center for *Imagery!* adventures. All new adventures received are carefully tested and then made available for all *Imagery!* players. Each issue of *The Image* will review the latest and best and adventures so you'll which ones to check out!

Only the adventures distributed by *Imagination* are considered official. Official adventures are supported with updates and bug fixes, while unofficial adventures may have problems.

We encourage you to send in your adventures so that others will be able to play them. Here's how:

- 1) Carefully test the adventure on your own. An adventure with a lot of bugs is no fun to play-- and it will take longer for us to ready it for distribution.
- 2) Write out a short description of the adventure. Include:
  - a) the object of the adventure
  - b) the time period (medieval, modern, etc.)
  - c) the level character necessary to play it
- 3) Describe any new commands and/or modifications you have made to the base adventure program. Include which lines of code have been changed.
- 4) Send the completed adventure, along with your description, to:

Imagination  
P.O. Box 190  
St. Paul, IN 47272

We'll test your adventure as soon as we receive it. Good luck!

All adventures are \$3.00 (\$5.00 for non-members) to cover postage. If you decide to distribute your adventures yourself, please don't charge any more than we do.

## Appendix F: Sample Libraries

This appendix contains tables showing the contents of the three monster/item libraries on the designer disk. The three libraries represent three common eras for *Imagery!* adventures: "medieval", "modern", and "futuristic". Careful study of the libraries will make designing your own monsters and items easier, as these samples can be a base on which to build.

### Abbreviations

The following abbreviations are used in the Monster tables:

Str = Strength  
Dex = Dexterity  
Frn % = Friendliness  
Cou % = Courage  
HP = Hit Points  
AB = Armour Bonus  
WB = Weapon Bonus  
Reac = Reaction

The following abbreviations are used in the Item tables:

HP = Hit Points  
Wt = Weight  
DB = Defense Bonus  
AB = Attack Bonus  
DD = Dice of Damage  
S/D = Sides per Die  
Dur = Duration  
Chgs = Charges

Remember: each library can hold a maximum of 15 monsters and 15 items. If you put together a library you would like to share with other *Imagery!* designers, send it in to *Imagination* along with an adventure or two. See Appendix E for further information on submitting *Imagery!* adventures.



**Modern Monsters**

<u>Name</u>	<u>Attack Phrase</u>	<u>Str</u>	<u>Dex</u>	<u>Frn %</u>	<u>Cou %</u>	<u>HP</u>	<u>AB</u>	<u>WB</u>	<u>Reac</u>	<u>Status</u>
creepy spider	Biting exposed limbs	2	16	0	20	6	5	10	0	normal
sewer rat	Biting and scratching	5	12	10	35	10	7	10	0	normal
mad dog	Snapping its foaming jaws	15	15	0	85	23	14	14	-1	berserk
teenage punk	Throwing a punch	17	16	25	65	22	15	15	0	normal
assassin	Kicking and punching	20	19	10	90	35	15	15	0	normal
gangster	Throwing punches left and right	21	15	27	55	30	15	15	0	normal
demon priest	Screaming and kicking	16	15	10	80	30	12	10	0	normal
killer pimp	Jabbing with his big fists	18	17	15	60	27	15	15	0	normal
escaped convict	Parrying and thrusting	21	16	5	77	31	16	17	0	normal
lone wolf	Baring its sharp fangs	16	16	12	60	30	17	17	0	normal

**Modern Items**

<u>Name</u>	<u>Effect</u>	<u>Value</u>	<u>HP</u>	<u>Wt</u>	<u>Type</u>	<u>DB</u>	<u>AB</u>	<u>DD</u>	<u>S/D</u>	<u>Power</u>	<u>Dur</u>	<u>Chgs</u>	<u>Light</u>
machine gun	Bullets fly from the barrel	600	100	20	weapon		15	3	8				none
44 magnum	*** Blam ***	475	85	12	weapon		15	2	8				none
pistol	*** Blam ***	350	100	10	weapon		15	2	6				none
nunchakus	Twirling the chucks	70	20	10	weapon		12	1	8				none
butterfly knife	Stabbing with the knife	75	20	5	weapon		12	1	6				none
flak jacket	<i>na</i>	700	500	35	armour	25							<i>na</i>
bullet-proof vest	<i>na</i>	1000	800	40	armour	35							<i>na</i>
first aid kit	You use the supplies and	75	15	7	magic					regeneration	5	5	<i>na</i>
dark liquid	You drink the dark liquid and	20	8	4	magic					poison	6	1	<i>na</i>
mysterious charm	You put the charm around your neck and	200	12	1	magic					cursing	50	666	<i>na</i>

### Futuristic Monsters

Name	Attack Phrase	Str	Dex	Frn %	Cou %	HP	AB	WB	Reac	Status
renegade	Moving fast and low	20	20	40	70	100	20	20	0	normal
giant squirrel	Gnawing and biting	18	18	25	30	85	25	25	0	normal
bounty hunter	Punching skillfully	20	19	50	90	150	20	20	0	normal
half-dead man	Swinging wildly	12	7	0	20	30	10	7	-1	normal
mutated thug	Hitting low	20	20	0	70	100	20	20	-1	normal
mind killer	Reaching with its thoughts	25	13	0	80	100	15	25	-1	normal
security robot	Pivoting instantly	35	30	0	100	250	35	30	-1	normal
carion eater	Snapping its beak	16	12	0	40	70	15	15	0	normal
skrayt	Waving its tentacles	19	16	0	60	125	30	17	-1	normal
hanging moss	Enveloping its prey	10	5	0	20	30	8	12	0	normal
life sapper	Stretching out its pincers	21	17	0	75	150	32	32	-1	normal

### Futuristic Items

Name	Effect	Value	HP	WI	Type	DB	AB	DD	S/D	Power	Dur	Chgs	Light
plasma autocannon	It fires a blast of hydrogen	3000	300	100	weapon		50	15	10				none
laser rifle	A thin beam shoots forth	2000	200	75	weapon		45	10	10				none
blaster rifle	It fires a ball of energy	2500	250	87	weapon		47	12	10				none
laser pistol	A thin beam shoots forth	1000	100	25	weapon		35	6	10				none
electro-neuron blade	Swinging the buzzing blade	1000	100	25	weapon		30	5	8				none
kevlar body armour	<i>na</i>	1700	550	55	armour	35							<i>na</i>
retro-reflect jacket	<i>na</i>	2000	600	70	armour	40							<i>na</i>
neuro-action armour	<i>na</i>	2700	700	100	armour	50							<i>na</i>
identity card	<i>na</i>	675	1	1	treasure								<i>na</i>
medi-quick kit	You activate the kit and	900	12	12	magic					regeneration	10	10	<i>na</i>