

A Guide to the Mazes of Menace

(Guidebook for SLASH'EM 0.0.3)

Eric S. Raymond

(Extensively edited and expanded for 3.0 by Mike Threepoint)

(Revised for SLASH'EM 0.0.3 by Warren Cheung)

1. Introduction

Having exhausted your own meager financial resources, as well as those of your parents, you find that you must end your formal education. Your lack of experience and skills leaves you facing a pretty grim future. You could look for some sort of menial job and hope to perform well enough to be noticed and perhaps rise in responsibilities until you were earning enough money to be comfortable. Or you could set out into the world and make your livelihood by prospecting, stealing, crusading, or just plain killing, for your gold. Over the objections of your local guildmaster, you opt to follow the adventuring route. After all, when adventurers came back this way they usually seemed better off than when they passed through the first time. And who was to say that all of those who did not return had not just kept going?

Asking around, you hear about a bauble, called the Amulet of Yendor by some, which, if you can find it, will bring you great wealth. One legend you were told even mentioned that the one who finds the amulet will be granted immortality by the gods. The amulet is rumored to be somewhere beyond the Valley of Gehennom, deep within the Mazes of Menace. You decide that even if the rumors of the amulet's powers are untrue, and even if it won't cure the common plague, you should at least be able to sell the tales of your adventures to the local minstrels for a tidy sum. You spend one last night fortifying yourself at the local inn, becoming more and more depressed as you watch the odds of your success being posted on the inn's walls getting lower and lower. In the morning you awake, gather together your belongings, and set off on your adventure...

Your abilities and strengths for dealing with the hazards of adventure will vary with your background and training.

Archeologists understand dungeons pretty well; this enables them to move quickly and sneak up on the local nasties. They start equipped with the tools for a proper scientific expedition.

Barbarians are warriors out of the hinterland, hardened to battle. They begin their quests with naught but uncommon strength, a trusty hauberk, and a great two-handed sword.

Cavemen and *Cavewomen* start with exceptional strength but, unfortunately, with neolithic weapons.

Doppelgangers have the envious ability to change form at will, at a cost of some mystic energy (*mana*), although what they become may be a bit of a surprise, even for them.

Elves are agile, quick, and sensitive; very little of what goes on will escape an Elf. The quality of Elven craftsmanship often gives them an advantage in arms and armor.

Flame Mages have managed to harness mystical energies into the control of the element of fire. Notwithstanding their pet dragons, woe be unto anyone who stands in the way of a skilled mage casting a fireball.

Gnomes are quite comfortable with the dank caverns of the dungeons and come well prepared, as they know what to expect. There is even rumor that a large number of their brethren work in the dungeons and call it their home.

Healers are wise in medicine and apothecary. They know the herbs and simples that can restore vitality, ease pain, anesthetize, and neutralize poisons; and with their instruments, they can divine a being's state of health or sickness. Their medical practice earns them quite reasonable amounts of money, with which they enter the dungeon.

Ice Mages command the forces of cold. An experienced Mage can summon great blizzards yet remain unaffected by the turmoil of the elements.

Knights are distinguished from the common skirmisher by their devotion to the ideals of chivalry and by the surpassing excellence of their armor.

Lycanthropes are wild beasts who draw their strength from the phases of the moon, and can transform into wolves when they channel their magical energies. Even unarmed, a Lycanthrope is a savage fighter, as many scarred by their deadly claws can attest.

Monks have mastered both the physical and the spiritual with their minds. Spurning the weapons of the world, an advanced Monk can destroy even the greatest of enemies with naught but his bare hands.

Necromancers have delved into the darkest of the magical lore, and mastered some of the most forbidden of the magical lore. Many have fallen to the armies of the undead that they are capable of bringing forth and controlling.

Priests and *Priestesses* are clerics militant, crusaders advancing the cause of righteousness with arms, armor, and arts thaumaturgic. Their ability to commune with deities via prayer occasionally extricates them from peril, but can also put them in it.

Rogues are agile and stealthy thieves, with knowledge of locks, traps, and poisons. Their advantage lies in surprise, which they employ to great advantage.

Samurai are the elite warriors of feudal Nippon. They are lightly armored and quick, and wear the *dai-sho*, two swords of the deadliest keenness.

Tourists start out with lots of gold (suitable for shopping with), a credit card, lots of food, some maps, and an expensive camera. Most monsters don't like being photographed.

Undead Slayers are specialists, trained to hunt the undead as well as other incarnations of evil. They are well aware of the weaknesses of their foes and come prepared. Few denizens of darkness ever encounter such warriors of light and live to tell of it.

Valkyries are hardy warrior women. Their upbringing in the harsh Northlands makes them strong, inures them to extremes of cold, and instills in them stealth and cunning.

Wizards start out with a knowledge of magic, a selection of magical items, and a particular affinity for dweomercraft. Although seemingly weak and easy to overcome at first sight, an experienced Wizard is a deadly foe.

You set out for the dungeon and after several days of uneventful travel you see the ancient ruins that mark the entrance to the Mazes of Menace. It is late at night, so you make camp at the entrance and spend the night sleeping under the open skies. In the morning, you gather your gear, eat what may be your last meal outside, and enter the dungeon.

2. What is going on here?

You have just begun a game of SLASH'EM. Your goal is to grab as much treasure as you can, retrieve the Amulet of Yendor, and escape the Mazes of Menace alive. On the screen is kept a map of where you have been and what you have seen on the current dungeon level; as you explore more of the level, it appears on the screen in front of you.

When SLASH'EM's ancestor *rogue* first appeared, its screen orientation was almost unique among computer fantasy games. Since then, screen orientation has become the norm rather than the exception; SLASH'EM continues this fine tradition. Unlike text adventure games that accept commands in pseudo-English sentences and explain the results in words, SLASH'EM commands are all one or two keystrokes and the results are displayed graphically on the screen. A minimum screen size of 24 lines by 80 columns is recommended; if the screen is larger, only a 21x80 section will be used for the map.

SLASH'EM generates a new dungeon every time you play it; even the authors still find it an entertaining and exciting game despite having won several times.

3. What do all those things on the screen mean?

SLASH'EM offers a variety of display options. The options available to you will vary from port to port, depending on the capabilities of your hardware and software, and whether various compile-time options were enabled when your executable was created. The three possible display options are: a monochrome character interface, a color character interface, and a graphical interface using small pictures called tiles. The two character interfaces allow fonts with other characters to be substituted, but the default assignments use standard ASCII characters to represent everything. There is no difference between the various display options with respect to game play. Because we cannot reproduce the tiles or colors in the Guidebook, and because it is common to all ports, we will use the default ASCII characters from the monochrome character display when referring to things you might see on the screen during your game.

In order to understand what is going on in SLASH'EM, first you must understand what SLASH'EM is doing with the screen. The SLASH'EM screen replaces the “You see ...” descriptions of text adventure games. Figure 1 is a sample of what a SLASH'EM screen might look like.

```

The bat bites!

-----
|...|  -----
|<.|####...@...$.|
|...-# |...B....+
|...|  |.d.....|
-----  -----|--

Player the Rambler      St:12 Dx:7 Co:18 In:11 Wi:9 Ch:15 Neutral
Dlv:1 $:0 HP:9(12) Pw:3(3) AC:10 Exp:1/19 T:257 Weak

```

Figure 1

3.1. The status lines (bottom)

The bottom two lines of the screen contain several cryptic pieces of information describing your current status. If either status line becomes longer than the width of the screen, you might not see all of it. Here are explanations of what the various status items mean (though your configuration may not have all the status items listed below):

Rank

Your character's name and professional ranking (based on the experience level, see below).

Strength

A measure of your character's strength; one of your six basic attributes. Your attributes can range from 3 to 18 inclusive (occasionally you may get super-strengths of the form 18/xx). The higher your strength, the stronger you are. Strength affects how successfully you perform physical tasks, how much damage you do in combat, and how much loot you can carry.

Dexterity

Dexterity affects your chances to hit in combat, to avoid traps, and do other tasks requiring agility or manipulation of objects.

Constitution

Constitution affects your ability to recover from injuries and other strains on your stamina.

Intelligence

Intelligence affects your ability to cast spells and read spellbooks.

Wisdom

Wisdom comes from your practical experience (especially when dealing with magic). It affects your magical energy.

Charisma

Charisma affects how certain creatures react toward you. In particular, it can affect the prices shopkeepers offer you.

Alignment

Lawful, Neutral, or Chaotic. Often, Lawful is good and Chaotic is evil. Your alignment influences how other monsters react toward you. Monsters of a like alignment are more likely to be non-aggressive, while those of an opposing alignment are more likely to be seriously offended at your presence.

Dungeon Level

How deep you are in the dungeon. You start at level one and the number increases as you go deeper into the dungeon. Some levels are special, and are identified by a name and not a number. The Amulet of Yendor is reputed to be somewhere beneath the twentieth level.

Gold

The number of gold pieces you are openly carrying. Gold which you have concealed in containers is not counted.

Hit Points

Your current and maximum hit points. Hit points indicate how much damage you can take before you die. The more you get hit in a fight, the lower they get. You can regain hit points by resting, or by using certain magical items or spells. The number in parentheses is the maximum number your hit points can reach.

Power

Spell points. This tells you how much mystic energy (*mana*) you have available for spell casting. Again, resting will regenerate the amount available.

Armor Class

A measure of how effectively your armor stops blows from unfriendly creatures. The lower this number is, the more effective the armor; it is quite possible to have negative armor class.

Experience

Your current experience level and experience points. As you adventure, you gain experience points. At certain experience point totals, you gain an experience level. The more experienced you are, the better you fight and withstand magical attacks. Many dungeons show only your experience level here.

Time

The number of turns elapsed so far, displayed if you have the **time** option set.

Hunger status

Your current hunger status, ranging from **Satiated** down to **Fainting**. If your hunger status is normal, it is not displayed.

Additional status flags may appear after the hunger status: **Conf** when you're confused, **FoodPois** or **Ill** when sick, **Blind** when you can't see, **Stun** when stunned, and **Hallu** when hallucinating.

3.2. The message line (top)

The top line of the screen is reserved for messages that describe things that are impossible to represent visually. If you see a "--More--" on the top line, this means that SLASH'EM has another message to display on the screen, but it wants to make certain that you've read the one that is there first. To read the next message, just press the space bar.

3.3. The map (rest of the screen)

The rest of the screen is the map of the level as you have explored it so far. Each symbol on the screen represents something. You can set various graphics options to change some of the symbols the game uses; otherwise, the game will use default symbols. Here is a list of what the default symbols mean:

- and	The walls of a room, or an open door.
.	The floor of a room, ice, or a doorless doorway.
#	A corridor, or possibly a kitchen sink (if your dungeon has sinks) or drawbridge.
<	A way to the previous level.
>	A way to the next level.
+	A closed door, or a spell book containing a spell you can learn.
@	A human (you, usually).
\$	A pile of gold.
^	A trap (once you detect it).
)	A weapon.
[A suit or piece of armor.
%	Something edible (not necessarily healthy).
?	A scroll.
/	A wand.
=	A ring.
!	A potion.
(A useful item (pick-axe, key, lamp...).
"	An amulet (or a spider web).
*	A gem or rock (possibly valuable, possibly worthless).
'	A boulder or statue.
0	An iron ball.
_	An altar, or an iron chain.
}	A pool of water or moat or a pool of lava.
{	A fountain.
\	An opulent throne.

a-zA-Z and other symbols

Letters and certain other symbols represent the various inhabitants of the Mazes of Menace. Watch out, they can be nasty and vicious. Sometimes, however, they can be helpful.

You need not memorize all these symbols; you can ask the game what any symbol represents with the '/' command (see the Commands section for more info).

4. Commands

Commands are initiated by typing one or two characters. Some commands, like "search", do not require that any more information be collected by SLASH'EM. Other commands might require additional information, for example a direction, or an object to be used. For those commands that require additional information, SLASH'EM will present you with either a menu of choices or with a command line prompt requesting information. Which you are presented with will depend chiefly on how you have set the **menustyle** option.

For example, a common question, in the form "What do you want to use? [a-zA-Z ?*]", asks you to choose an object you are carrying. Here, "a-zA-Z" are the inventory letters of your possible choices. Typing '?' gives you an inventory list of these items, so you can see what each letter refers to. In this example, there is also a '*' indicating that you may choose an object not on the list, if you wanted to use something unexpected. Typing a '*' lists your entire inventory, so you can see the inventory letters of every object you're carrying. Finally, if you change your mind and decide you don't want to do this command after all, you can press the ESC key to abort the command.

You can put a number before most commands to repeat them that many times; for example, “10s” will search ten times. If you have the **number_pad** option set, you must type ‘n’ to prefix a count, so the example above would be typed “n10s” instead. Commands for which counts make no sense ignore them. In addition, movement commands can be prefixed for greater control (see below). To cancel a count or a prefix, press the ESC key.

The list of commands is rather long, but it can be read at any time during the game through the ‘?’ command, which accesses a menu of helpful texts. As well, there is now a menusystem available through the ‘’ command for those who would rather page through menus than hunt and peck for keys. Here are the commands for your reference:

- ? Help menu: display one of several help texts available.
- ‘ Main menu: access the menusystem.
- / Tell what a symbol represents. You may choose to specify a location or type a symbol (or even a whole word) to define. If the **help** option is on, and SLASH'EM has some special information about an object or monster that you looked at, you'll be asked if you want “More info?”. If **help** is off, then you'll only get the special information if you explicitly ask for it by typing in the name of the monster or object.
- & Tell what a command does.
- < Go up to the previous level (if you are on the staircase or ladder).
- > Go down to the next level (if you are on the staircase or ladder).

[yuhjklbn]

Go one step in the direction indicated (see Figure 2). If there is a monster there, you will fight the monster instead. Only these one-step movement commands cause you to fight monsters; the others (below) are “safe.”

y k u	7 8 9
\ /	\ /
h- . -l	4- . -6
/ \	/ \
b j n	1 2 3

(if **number_pad** is set)

Figure 2

[YUHJKLBN]

Go in that direction until you hit a wall or run into something.

m[yuhjklbn]

Prefix: move without picking up any objects.

M[yuhjklbn]

Prefix: move far, no pickup.

g[yuhjklbn]

Prefix: move until something interesting is found.

G[yuhjklbn] or <CONTROL->[yuhjklbn]

Prefix: same as ‘g’, but forking of corridors is not considered interesting.

. Rest, do nothing for one turn.

a Apply (use) a tool (pick-axe, key, lamp...).

A Remove one or more worn items, such as armor. Use ‘T’ (take off) to take off only one piece of armor or ‘R’ (remove) to take off only one accessory.

^A Redo the previous command.

c Close a door.

C Call (name) an individual monster.

^C Panic button. Quit the game.

d Drop something. Ex. “d7a” means drop seven items of object *a*.

- D Drop several things. In answer to the question “What kinds of things do you want to drop? [!= au]” you should type zero or more object symbols possibly followed by ‘a’ and/or ‘u’.
- Da - drop all objects, without asking for confirmation.
 - Du - drop only unpaid objects (when in a shop).
 - D%u - drop only unpaid food.
- ^D Kick something (usually a door).
- e Eat food.
- E Engrave a message on the floor. Engraving the word “Elbereth” will cause most monsters to not attack you hand-to-hand (but if you attack, you will rub it out); this is often useful to give yourself a breather. (This feature may be compiled out of the game, so your version might not have it.)
- E- - write in the dust with your fingers.
- f Fire ammunition from quiver.
- i List your inventory (everything you’re carrying).
- I List selected parts of your inventory.
- I* - list all gems in inventory;
 - Iu - list all unpaid items;
 - Ix - list all used up items that are on your shopping bill;
 - I\$ - count your money.
- o Open a door.
- O Set options. You will be asked to enter an option line. If you enter a blank line, the current options are reported. Entering ‘?’ will get you explanations of the various options. Otherwise, you should enter a list of options separated by commas. The available options are listed later in this Guidebook. Options are usually set before the game, not with the ‘O’ command; see the section on options below.
- p Pay your shopping bill/Shopkeeper services.
- P Put on a ring or other accessory (amulet, blindfold).
- ^P Repeat previous message (subsequent ^P’s repeat earlier messages).
- q Quaff (drink) a potion.
- Q Ready ammunition in quiver.
- r Read a scroll or spell book.
- R Remove an accessory (ring, amulet, etc).
- ^R Redraw the screen.
- s Search for secret doors and traps around you. It usually takes several tries to find something.
- S Save the game. The game will be restored automatically the next time you play.
- t Throw an object or shoot a projectile.
- T Take off armor.
- ^T Teleport, if you have the ability.
- v Display version number.
- V Display the game history.
- w Wield weapon. w- means wield nothing, use your bare hands.
- W Wear armor.
- x Switch weapon slots.
- X Enter explore (discovery) mode.
- z Zap a wand.

- Z Zap (cast) a spell.
- ^Z Suspend the game (UNIX[®] versions with job control only).
- : Look at what is here.
- ; Show what type of thing a visible symbol corresponds to.
- , Pick up some things.
- @ Toggle the **autopickup** option on and off.
- ^ Ask for the type of a trap you found earlier.
-) Tell what weapon you are wielding.
- [Tell what armor you are wearing.
- = Tell what rings you are wearing.
- " Tell what amulet you are wearing.
- (Tell what tools you are using.
- * List and change items currently in use.
- \$ Count your gold pieces.
- + List the spells you know.
- \ Show what types of objects have been discovered.
- ! Escape to a shell.
- # Perform an extended command. As you can see, the authors of SLASH'EM used up all the letters, so this is a way to introduce the less useful commands, or commands used under limited circumstances. You may obtain a list of them by entering '?'. What extended commands are available depends on what features the game was compiled with.

If your keyboard has a meta key (which, when pressed in combination with another key, modifies it by setting the 'meta' [8th, or 'high'] bit), you can invoke the extended commands by meta-ing the first letter of the command. In OS/2, PC, and ST SLASH'EM, the 'Alt' key can be used in this fashion; on the Amiga set the **altmeta** option to get this behavior.

- M-a Adjust inventory letters (most useful when the **fixinv** option is "on").
- M-b Steal
- M-c Talk to someone.
- M-d Dip an object into something.
- M-e Advance or check weapons skills.
- M-f Force a lock.
- M-i Invoke an object's special powers.
- M-j Jump to another location.
- M-k Advance or check weapons skills. Same as "#enhance"
- M-l Loot a box on the floor.
- M-m Use a monster's special ability.
- M-n Name an item or type of object.
- M-o Offer a sacrifice to the gods.
- M-p Pray to the gods for help.
- M-q Quit the game.

[®]UNIX is a registered trademark of AT&T.

- M-r Rub a lamp.
- M-s Sit down.
- M-t Use a class specific technique.
- M-u Untrap something (trap, door, or chest).
- M-v Print compile time options for this version of SLASH'EM.
- M-w Wipe off your face.
- M-y Polymorph yourself.

If the **number_pad** option is on, some additional letter commands are available:

- j Jump to another location. Same as “#jump” or “M-j”.
- k Kick something (usually a door). Same as ‘^D’.
- l Loot a box on the floor. Same as “#loot” or “M-l”.
- N Name an item or type of object. Same as “#name” or “M-N”.
- u Untrap a trap, door, or chest. Same as “#untrap” or “M-u”.

5. Rooms and corridors

Rooms and corridors in the dungeon are either lit or dark. Any lit areas within your line of sight will be displayed; dark areas are only displayed if they are within one space of you. Walls and corridors remain on the map as you explore them.

Secret corridors are hidden. You can find them with the ‘s’ (search) command.

5.1. Doorways

Doorways connect rooms and corridors. Some doorways have no doors; you can walk right through. Others have doors in them, which may be open, closed, or locked. To open a closed door, use the ‘o’ (open) command; to close it again, use the ‘c’ (close) command.

You can get through a locked door by using a tool to pick the lock with the ‘a’ (apply) command, or by kicking it open with the ‘^D’ (kick) command.

Open doors cannot be entered diagonally; you must approach them straight on, horizontally or vertically. Doorways without doors are not restricted in this fashion.

Doors can be useful for shutting out monsters. Most monsters cannot open doors, although a few don’t need to (ex. ghosts can walk through doors).

Secret doors are hidden. You can find them with the ‘s’ (search) command. Once found they are in all ways equivalent to normal doors.

5.2. Traps (^)

There are traps throughout the dungeon to snare the unwary delver. For example, you may suddenly fall into a pit and be stuck for a few turns trying to climb out. Traps don’t appear on your map until you see one triggered by moving onto it, see something fall into it, or you discover it with the ‘s’ (search) command. Monsters can fall prey to traps, too, which can be used as a useful defensive strategy.

6. Monsters

Monsters you cannot see are not displayed on the screen. Beware! You may suddenly come upon one in a dark place. Some magic items can help you locate them before they locate you (which some monsters can do very well).

6.1. Fighting

If you see a monster and you wish to fight it, just attempt to walk into it. Many monsters you find will mind their own business unless you attack them. Some of them are very dangerous when angered. Remember: discretion is the better part of valor.

6.2. Your pet

You start the game with a little dog ('d') or cat ('f'), which follows you about the dungeon and fights monsters with you. Like you, your pet needs food to survive. It usually feeds itself on fresh carrion and other meats. If you're worried about it or want to train it, you can feed it, too, by throwing it food. A properly trained pet can be very useful under certain circumstances.

Your pet also gains experience from killing monsters, and can grow over time, gaining hit points and doing more damage. Initially, your pet may even be better at killing things than you, which makes pets useful for low-level characters.

Your pet will follow you up and down staircases if it is next to you when you move. Otherwise your pet will be stranded and may become wild.

6.3. Bones levels

You may encounter the shades and corpses of other adventurers (or even former incarnations of yourself!) and their personal effects. Ghosts are hard to kill, but easy to avoid, since they're slow and do little damage. You can plunder the deceased adventurer's possessions; however, they are likely to be cursed. Beware of whatever killed the former player; it is probably still lurking around, gloating over its last victory.

7. Objects

When you find something in the dungeon, it is common to want to pick it up. In SLASH'EM, this is accomplished automatically by walking over the object (unless you turn off the **autopickup** option (see below), or move with the 'm' prefix (see above)), or manually by using the ';' command.

If you're carrying too many things, SLASH'EM will tell you so and you won't be able to pick up anything more. Otherwise, it will add the object(s) to your pack and tell you what you just picked up.

When you pick up an object, it is assigned an inventory letter. Many commands that operate on objects must ask you to find out which object you want to use. When SLASH'EM asks you to choose a particular object you are carrying, you are usually presented with a list of inventory letters to choose from (see Commands, above).

Some objects, such as weapons, are easily differentiated. Others, like scrolls and potions, are given descriptions which vary according to type. During a game, any two objects with the same description are the same type. However, the descriptions will vary from game to game.

When you use one of these objects, if its effect is obvious, SLASH'EM will remember what it is for you. If its effect isn't extremely obvious, you will be asked what you want to call this type of object so you will recognize it later. You can also use the "#name" command for the same purpose at any time, to name all objects of a particular type or just an individual object.

7.1. Curses and Blessings

Any object that you find may be cursed, even if the object is otherwise helpful. The most common effect of a curse is being stuck with (and to) the item. Cursed weapons weld themselves to your hand when wielded, so you cannot unwield them. Any cursed item you wear is not removable by ordinary means. In addition, cursed arms and armor usually, but not always, bear negative enchantments that make them less effective in combat. Other cursed objects may act poorly or detrimentally in other ways.

Objects can also be blessed. Blessed items usually work better or more beneficially than normal uncursed items. For example, a blessed weapon will do more damage against demons.

There are magical means of bestowing or removing curses upon objects, so even if you are stuck with one, you can still have the curse lifted and the item removed. Priests and Priestesses have an innate sensitivity to this

property in any object, so they can more easily avoid cursed objects than other character classes.

An item with unknown status will be reported in your inventory with no prefix. An item which you know the state of will be distinguished in your inventory by the presence of the word “cursed”, “uncursed” or “blessed” in the description of the item.

7.2. Weapons (‘’)

Given a chance, most monsters in the Mazes of Menace will gratuitously try to kill you. You need weapons for self-defense (killing them first). Without a weapon, you do only 1-2 hit points of damage (plus bonuses, if any).

There are wielded weapons, like maces and swords, and thrown weapons, like arrows and spears. To hit monsters with a weapon, you must wield it and attack them, or throw it at them. You can simply select to throw a spear.

To shoot an arrow, you should first wield a bow, then throw the arrow. An alternative method would be wield a bow and place the arrows in your Quiver, after which you fire the ammunition. Crossbows shoot crossbow bolts. Slings hurl rocks and (other) stones (like gems).

You can wield only one weapon at a time, but you can change weapons unless you’re wielding a cursed one. Exchanging hands allows you to ready both a primary and secondary weapon and swap between them at the touch of a key.

To switch to bare hands, wield ‘-’, or use the ‘A’ command which allows you to unwield the current weapon in addition to taking off other worn items.

The “#enhance” command will be present if the “weapon skills” feature is enabled, and deals with your proficiency in various types of weapons (as well as spells). Skilled adventurers find themselves much more capable with their weapons.

Enchanted weapons have a “plus” (or “to hit enhancement” which can be either positive or negative) that adds or subtracts to/from your chance to hit and the damage you do to a monster. The only way to determine a weapon’s enchantment is to have it magically identified somehow.

Most weapons are subject to some type of damage like rust. Such damage can be repaired.

Those of you in the audience who are AD&D players, be aware that each weapon which exists in AD&D does the same damage to monsters in SLASH'EM. Some of the more obscure weapons (such as the *aklys*, *lucern hammer*, and *bec-de-corbin*) are defined in an appendix to *Unearthed Arcana*, an AD&D supplement.

The commands to use weapons are ‘w’ (wield), ‘t’ (throw), and the “#enhance” extended command.

7.3. Armor (‘[’)

Lots of unfriendly things lurk about; you need armor to protect yourself from their blows. Some types of armor offer better protection than others. Your armor class is a measure of this protection. Armor class (AC) is measured as in AD&D, with 10 being the equivalent of no armor, and lower numbers meaning better armor. Each suit of armor which exists in AD&D gives the same protection in SLASH'EM. Here is an (incomplete) list of the armor classes provided by various suits of armor:

dragon scale mail	1
crystal plate mail	3
plate mail	3
bronze plate mail	4
splint mail	4
banded mail	4
elven mithril-coat	5
chain mail	5
scale mail	6
ring mail	7
studded leather armor	7
leather armor	8

no armor

10

You can also wear other pieces of armor (ex. helmets, boots, shields, cloaks) to lower your armor class even further, but you can only wear one item of each category (one suit of armor, one cloak, one helmet, one shield, and so on) at a time.

If a piece of armor is enchanted, its armor protection will be better (or worse) than normal, and its “plus” (or minus) will subtract from your armor class. For example, a +1 chain mail would give you better protection than normal chain mail, lowering your armor class one unit further to 4. When you put on a piece of armor, you immediately find out the armor class and any “plusses” it provides. Cursed pieces of armor usually have negative enchantments (minuses) in addition to being unremovable.

Many types of armor are subject to some kind of damage like rust. Such damage can be repaired. Some types of armor may inhibit spell casting.

The commands to use armor are ‘W’ (wear) and ‘T’ (take off). The ‘A’ command can also be used to take off armor as well as other worn items.

7.4. Food (‘%’)

Food is necessary to survive. If you go too long without eating you will faint, and eventually die of starvation. Some types of food will spoil, and become unhealthy to eat, if not protected. Food stored in ice boxes or tins (“cans”) will usually stay fresh, but ice boxes are heavy, and tins take a while to open.

When you kill monsters, they usually leave corpses which are also “food.” Many, but not all, of these are edible; some also give you special powers when you eat them. A good rule of thumb is “you are what you eat.”

You can name one food item after something you like to eat with the **fruit** option, if your dungeon has it.

The command to eat food is ‘e’.

7.5. Scrolls (‘?’)

Scrolls are labeled with various titles, probably chosen by ancient wizards for their amusement value (ex. “READ ME,” or “HOLY BIBLE” backwards). Scrolls disappear after you read them (except for blank ones, without magic spells on them).

One of the most useful of these is the *scroll of identify*, which can be used to determine what another object is, whether it is cursed or blessed, and how many uses it has left. Some objects of subtle enchantment are difficult to identify without these.

A mail daemon may run up and deliver mail to you as a *scroll of mail* (on versions compiled with this feature). To use this feature on versions where SLASH'EM mail delivery is triggered by electronic mail appearing in your system mailbox, you must let SLASH'EM know where to look for new mail by setting the “MAIL” environment variable to the file name of your mailbox. You may also want to set the “MAILREADER” environment variable to the file name of your favorite reader, so SLASH'EM can shell to it when you read the scroll. On versions of SLASH'EM where mail is randomly generated internal to the game, these environment variables are ignored. You can disable the mail daemon by turning off the **mail** option.

The command to read a scroll is ‘r’.

7.6. Potions (‘!’)

Potions are distinguished by the color of the liquid inside the flask. They disappear after you quaff them.

Clear potions are potions of water. Sometimes these are blessed or cursed, resulting in holy or unholy water. Holy water is the bane of the undead, so potions of holy water are good things to throw (‘t’) at them. It is also sometimes very useful to dip (“#dip”) an object into a potion.

The command to drink a potion is ‘q’ (quaff).

7.7. Wands (‘/’)

Magic wands usually have multiple magical charges. Some wands are directional—you must give a direction in which to zap them. You can also zap them at yourself (just give a ‘.’ or ‘s’ for the direction). Be warned, however, for this is often unwise. Other wands are nondirectional—they don’t require a direction. The number of charges in a wand is random and decreases by one whenever you use it.

When the number of charges left in a wand becomes zero, attempts to use the wand will usually result in nothing happening. Occasionally, however, it may be possible to squeeze the last few mana points from an otherwise spent wand.

In a truly desperate situation, when your back is up against the wall, you might decide to go for broke and break your wand. This is not for the faint of heart. Doing so will almost certainly cause a catastrophic release of magical energies.

The command to use a wand is ‘z’ (zap). To break one, use the ‘a’ (apply) command.

7.8. Rings (‘=’)

Rings are very useful items, since they are relatively permanent magic, unlike the usually fleeting effects of potions, scrolls, and wands.

Putting on a ring activates its magic. You can wear only two rings, one on each ring finger.

Most rings also cause you to grow hungry more rapidly, the rate varying with the type of ring.

The commands to use rings are ‘P’ (put on) and ‘R’ (remove).

7.9. Spell books (‘+’)

Spell books are tomes of mighty magic. When studied with the ‘r’ (read) command, they bestow the knowledge of a spell—unless the attempt backfires. Reading a cursed spell book or one with mystic runes beyond your ken can be harmful to your health!

A spell (even when learned) can also backfire when you cast it. If you attempt to cast a spell well above your experience level, or cast it at a time when your luck is particularly bad, you can end up wasting both the energy and the time required in casting.

Casting a spell calls forth magical energies and focuses them with your naked mind. Releasing the magical energy releases some of your memory of the spell with it. Each time you cast a spell, your familiarity with it will dwindle, until you eventually forget the details completely and must relearn it.

Casting a spell also requires flexible movement, and wearing various types of armor may interfere with that.

The “#enhance” command will be present if the “weapon skills” feature is enabled, and deals with your proficiency in various types of spells (as well as weapons). Skilled adventurers can command awesome spells at their fingertips.

The command to read a spell book is the same as for scrolls, ‘r’ (read). The ‘+’ command lists your current spells and the number of spell points they require. The ‘Z’ (cast) command casts a spell. The “#enhance” extended command advances your spellcasting skills.

7.10. Tools (‘.’)

Tools are miscellaneous objects with various purposes. Some tools, like wands, have a limited number of uses. For example, lamps burn out after a while. Other tools are containers, which objects can be placed into or taken out of.

The command to use tools is ‘a’ (apply).

7.10.1. Chests and boxes

You may encounter chests or boxes in your travels. These can be opened with the “#loot” extended command when they are on the floor, or with the ‘a’ (apply) command when you are carrying one. However, chests are often locked, and are in any case unwieldy objects. You must set one down before unlocking it by kicking it, using a key

or lock-picking tool with the 'a' (apply) command, or by using a weapon to force the lock with the "#force" extended command.

Some chests are trapped, causing nasty things to happen when you unlock or open them. You can check for and try to deactivate traps with the "#untrap" extended command.

7.11. Amulets ("")

Amulets are very similar to rings, and often more powerful. Like rings, amulets have various magical properties, some beneficial, some harmful, which are activated by putting them on.

Only one amulet may be worn at a time, around your neck.

The commands to use amulets are the same as for rings, 'P' (put on) and 'R' (remove).

7.12. Gems ("*")

Some gems are valuable, and can be sold for a lot of gold. They are also a far more efficient way of carrying your riches. Valuable gems increase your score if you bring them with you when you exit.

Other small rocks are also categorized as gems, but they are much less valuable. All rocks, however, can be used as projectile weapons (if you have a sling). In the most desperate of cases, you can still throw them by hand.

7.13. Large rocks ("")

Statues and boulders are not particularly useful, and are generally heavy. It is rumored that some statues are not what they seem.

Very large humanoids (giants and their ilk) have been known to use boulders as weapons.

7.14. Gold ("'\$")

Gold adds to your score, and you can buy things in shops with it. Your version of SLASH'EM may display how much gold you have on the status line. If not, the '\$' command will count it. There are a number of monsters in the dungeon that may be influenced by the amount of gold you are carrying (shopkeepers aside).

8. Options

Due to variations in personal tastes and conceptions of how SLASH'EM should do things, there are options you can set to change how SLASH'EM behaves.

8.1. Setting the options

Options may be set in a number of ways. Within the game, the 'O' command allows you to view all options and change most of them. You can also set options automatically by placing them in the SLASHEMOPTIONS environment variable or in a configuration file. Some versions of SLASH'EM also have front-end programs that allow you to set options before starting the game.

8.2. Using the SLASHEMOPTIONS environment variable

The SLASHEMOPTIONS variable is a comma-separated list of initial values for the various options. Some can only be turned on or off. You turn one of these on by adding the name of the option to the list, and turn it off by typing a '!' or "no" before the name. Others take a character string as a value. You can set string options by typing the option name, a colon, and then the value of the string. The value is terminated by the next comma or the end of string.

For example, to set up an environment variable so that "female" is on, "autopickup" is off, the name is set to "Blue Meanie", and the fruit is set to "papaya", you would enter the command

```
% setenv SLASHEMOPTIONS "female,!autopickup,name:Blue Meanie,fruit:papaya"
```

in *csh* (note the need to escape the ! since it's special to the shell), or

```
$ SLASHEMOPTIONS="female,!autopickup,name:Blue Meanie,fruit:papaya"
$ export SLASHEMOPTIONS
```

in *sh* or *ksh*.

8.3. Using a configuration file

Any line in the configuration file starting with “OPTIONS=” may be filled out with options in the same syntax as in SLASHEMOPTIONS. Any line starting with “DUNGEON=”, “EFFECTS=”, “MONSTERS=”, “OBJECTS=”, or “TRAPS=” is taken as defining the corresponding **dungeon**, **effects**, **monsters**, **objects** or **traps** option in a different syntax, a sequence of decimal numbers giving the character position in the current font to be used in displaying each entry. Such a sequence can be continued to multiple lines by putting a ‘\’ at the end of each line to be continued. Any line starting with ‘#’ is treated as a comment.

The default name of the configuration file varies on different operating systems, but SLASHEMOPTIONS can also be set to the full name of a file you want to use (possibly preceded by an ‘@’).

8.4. Customization options

Here are explanations of what the various options do. Character strings that are too long may be truncated. Some of the options listed may be inactive in your dungeon.

autopickup

Pick up things you move onto by default (default on).

badgername

Name your starting badger (ex. “badgername:Jack”). Cannot be set with the ‘O’ command.

batname

Name your starting bat (ex. “batname:Drake”). Cannot be set with the ‘O’ command.

BIOS

Use BIOS calls to update the screen display quickly and to read the keyboard (allowing the use of arrow keys to move) on machines with an IBM PC compatible BIOS ROM (default off, OS/2, PC, and ST SLASH'EM only).

catname

Name your starting cat (ex. “catname:Tabby”). Cannot be set with the ‘O’ command.

checkpoint

Save game state after each level change, for possible recovery after program crash (default on).

color

Use color for different monsters, objects, and dungeon features (default on for microcomputers).

confirm

Have user confirm attacks on pets, shopkeepers, and other peaceable creatures (default on).

DECgraphics

Use a predefined selection of characters from the DEC VT-xxx/DEC Rainbow/ANSI line-drawing character set to display the dungeon/effects/traps instead of having to define a full graphics set yourself (default off). This option also sets up proper handling of graphics characters for such terminals, so you should specify it when appropriate even if you override the selections with your own graphics strings.

disclose

Offer to disclose various information when the game ends (default all). The possibilities are identifying your inventory (‘i’), disclosing your attributes (‘a’), summarizing monsters that have been vanquished (‘v’), and listing monster species that have been genocided (‘g’). Note that the vanquished monsters list includes all monsters killed by traps and each other as well as by you.

dogname

Name your starting dog (ex. “dogname:Fang”). Cannot be set with the ‘O’ command.

dungeon

Set the graphics symbols for displaying the dungeon (default “|-----||.-|++##<><>_||##{.}.## #”). The

dungeon option should be followed by a string of 1-38 characters to be used instead of the default map-drawing characters. The dungeon map will use the characters you specify instead of the default symbols, and default symbols for any you do not specify. Remember that you may need to escape some of these characters if, for example, you use *ash*.

Note that this option string is escape-processed in conventional C fashion. This means that ‘\’ is a prefix to take the following character literally, and not as a special prefix. The special escape form ‘\m’ switches on the meta bit in the following character, and the ‘^’ prefix causes the following character to be treated as a control character.

The order of the symbols is: solid rock, vertical wall, horizontal wall, upper left corner, upper right corner, lower left corner, lower right corner, cross wall, upward T wall, downward T wall, leftward T wall, rightward T wall, no door, vertical open door, horizontal open door, vertical closed door, horizontal closed door, floor of a room, dark corridor, lit corridor, stairs up, stairs down, ladder up, ladder down, altar, grave, throne, kitchen sink, toilet, fountain, pool or moat, ice, lava, vertical lowered drawbridge, horizontal lowered drawbridge, vertical raised drawbridge, horizontal raised drawbridge, air, cloud, under water.

You might want to use ‘+’ for the corners and T walls for a more aesthetic, boxier display. Note that in the next release, new symbols may be added, or the present ones rearranged.

Cannot be set with the ‘O’ command.

effects

Set the graphics symbols for displaying special effects (default “|-\/*!)(0#@*-/\\|\\-/-\\| \\-/-”). The **effects** option should be followed by a string of 1-29 characters to be used instead of the default special-effects characters. This string is subjected to the same processing as the **dungeon** option.

The order of the symbols is: vertical beam, horizontal beam, left slant, right slant, digging beam, camera flash beam, left boomerang, right boomerang, four glyphs giving the sequence for magic resistance displays; the eight surrounding glyphs for swallowed display; nine glyphs for explosions. An explosion consists of three rows (top, middle, and bottom) of three characters. The explosion is centered in the center of this 3 by 3 array.

Note that in the next release, new symbols may be added, or the present ones rearranged.

Cannot be set with the ‘O’ command.

female

Set your sex (default off). Cannot be set with the ‘O’ command.

fixinv An object’s inventory letter sticks to it when it’s dropped (default on). If this is off, dropping an object shifts all the remaining inventory letters.

fruit Name a fruit after something you enjoy eating (ex. “fruit:mango”) (default “slime mold”). Basically a nostalgic whimsy that SLASH’EM uses from time to time. You should set this to something you find more appealing than slime mold. Apples, oranges, pears, bananas, and melons already exist in SLASH’EM, so don’t use those.

ghoulname

Name your starting ghoul (ex. “ghoulname:Casper”). Cannot be set with the ‘O’ command.

help If more information is available for an object looked at with the ‘/’ command, ask if you want to see it (default on). Turning help off makes just looking at things faster, since you aren’t interrupted with the “More info?” prompt, but it also means that you might miss some interesting and/or important information.

hilite_pet

Visually distinguish pets from similar animals (default off). In text windowing, use text highlighting when color is turned off; with X tiles, display a heart symbol near pets.

IBMgraphics

Use a predefined selection of IBM extended ASCII characters to display the dungeon/effects/traps instead of having to define a full graphics set yourself (default off). This option also sets up proper handling of graphics characters for such terminals, so you should specify it when appropriate even if you override the selections with your own graphics strings.

ignintr

Ignore interrupt signals, including breaks (default off).

keep_savefile

Keeps the save file after restore. SLASH'EM usually deletes your savefile after you restore, making death permanent. This option can allow you to restore from the last save. (default off).

legacy

Display an introductory message when starting the game (default on).

lit_corridor

Show corridor squares seen by night vision or a light source held by your character as lit (default off).

mail Enable**male**

Set your sex (default on, most hackers are male). Cannot be set with the 'O' command.

menustyle

Controls the interface used when you need to choose various objects (in response to the Drop command, for instance). The value specified should be the first letter of one of the following: traditional, combination, partial, or full. Traditional was the only interface available for earlier versions; it consists of a prompt for object class characters, followed by an object-by-object prompt for all items matching the selected object class(es). Combination starts with a prompt for object class(es) of interest, but then displays a menu of matching objects rather than prompting one-by-one. Partial skips the object class filtering and immediately displays a menu of all objects. Full displays a menu of object classes rather than a character prompt, and then a menu of matching objects for selection.

menu_deselect_all

Menu character accelerator to deselect all items in a menu. Implemented by the X11 and tty ports. Default '~'.

menu_deselect_page

Menu character accelerator deselect all items on this page of a menu. Implemented only by the tty port. Default '\'.

menu_first_page

Menu character accelerator to jump to the first page in a menu. Implemented only by the tty port. Default '^'.

menu_invert_all

Menu character accelerator to invert all items in a menu. Implemented by the X11 and tty ports. Default '@'.

menu_invert_page

Menu character accelerator to invert all items on this page of a menu. Implemented only by the tty port. Default '~'.

menu_last_page

Menu character accelerator to jump to the last page in a menu. Implemented only by the tty port. Default '|'.

menu_next_page

Menu character accelerator to goto the next menu page. Implemented only by the tty port. Default '>'.

menu_previous_page

Menu character accelerator to goto the previous menu page. Implemented only by the tty port. Default '<'.

menu_search

Menu character accelerator to search for a menu item. Implemented only by the X11 port. Default ':'.

menu_select_all

Menu character accelerator to select all items in a menu. Implemented by the X11 and tty ports. Default '.'.

menu_select_page

Menu character accelerator to select all items on this page of a menu. Implemented only by the tty port. Default ','.

monsters

Set the characters used to display monster classes (default "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ@ '&;~]"). This string is subjected to the same processing as the **dungeon** option. The order of the symbols is ant or other insect, blob, cockatrice, dog or other canine, eye or sphere, feline, gremlin, humanoid, imp or minor demon, jelly, kobold, leprechaun, mimic, nymph, orc, piercer, quadruped, rodent, spider, trapper or lurker above, unicorn, vortex, worm, xan or other mythical/fantastic insect, light, zrutu, angelic being, bat, centaur, dragon, elemental, fungus or mold, gnome, giant humanoid, invisible stalker, jabberwock, Keystone Kop, lich, mummy, naga, ogre, pudding or ooze, quantum mechanic, rust monster, snake, troll, umber hulk, vampire, wraith, xorn, yeti or ape or other large beast, zombie, human, ghost, golem, demon, sea monster, lizard, long worm tail, and mimic. Cannot be set with the 'O' command.

msghistory

The number of top line messages to save (and recall with ^P) (default 20). Cannot be set with the 'O' command.

name

Set your character's name (defaults to your user name). You can also set your character class by appending a dash and the first letter of the character class (that is, by suffixing one of **-A -B -C -E -H -K -P -R -S -T -V -W**). If -@ is used for the class, then a random one will be automatically chosen. Cannot be set with the 'O' command.

news

Read the SLASH'EM news file, if present (default on). Since the news is shown at the beginning of the game, there's no point in setting this with the 'O' command.

null

Send padding nulls to the terminal (default off).

number_pad

Use the number keys to move instead of [yuhjklbn] (default off).

objects

Set the characters used to display object classes (default "[]=[%!+/\$*0_]"). This string is subjected to the same processing as the **dungeon** option. The order of the symbols is illegal-object (should never be seen), weapon, armor, ring, amulet, tool, food, potion, scroll, spell book, wand, gold, gem or rock, boulder or statue, iron ball, chain, and venom. Cannot be set with the 'O' command.

packorder

Specify the order to list object types in (default "[]=[%!+/\$*0_]"). The value of this option should be a string containing the symbols for the various object types.

perm_invent

If true, always display your current inventory in a window. This only makes sense for windowing system interfaces that implement this feature.

pettype

Specify the type of your initial pet, if you are playing a character class that uses both types of pets. Possible values are "cat" and "dog". Cannot be set with the 'O' command.

pickup_burden

Specify at what encumbrance level to begin prompting for confirmation when picking up objects. Default is moderate encumbrance.

pickup_types

Specify the object types to be picked up when **autopickup** is on. Default is all types.

preload_tiles

For the protected mode MSDOS version, control whether tiles get pre-loaded into RAM at the start of the game. Doing so enhances performance of the tile graphics, but uses more memory. (default on). Cannot be set with the 'O' command.

ratname

Name your starting rat (ex. "ratname:Sherwin"). Cannot be set with the 'O' command.

rawio

Force raw (non-cbreak) mode for faster output and more bulletproof input (MS-DOS sometimes treats '^P' as a printer toggle without it) (default off). Note: DEC Rainbows hang if this is turned on. Cannot be set with the 'O' command.

reddragonname

Name your starting red dragon (ex. "reddragonname:Fyre"). Cannot be set with the 'O' command.

rest_on_space

Make the space bar a synonym for the '.' (rest) command (default off).

safe_pet

Prevent you from (knowingly) attacking your pets (default on).

snakename

Name your starting snake (ex. "snakename:Joe"). Cannot be set with the 'O' command. **keep_savefile** is also on.

scores Control what parts of the score list you are shown at the end (ex. "scores:5 top scores/4 around my score/own scores"). Only the first letter of each category ('t', 'a', or 'o') is necessary.

showexp

Show your accumulated experience points on bottom line (default off).

showscore

Show your approximate accumulated score on bottom line (default off).

showdmg

Show damage inflicted/damage received (default off).

showweight

Show weight in inventory (default off).

silent Suppress**sortpack**

Sort the pack contents by type when displaying inventory (default on).

standout

Boldface monsters and "--More--" (default off).

time

Show the elapsed game time in turns on bottom line (default off).

timed_delay

When pausing momentarily for display effect, such as with explosions and moving objects, use a timer rather than sending extra characters to the screen. (Applies to "tty" interface only; "X11" interface always uses a timer based delay. The default is on if configured into the program.)

tombstone

Draw a tombstone graphic upon your death (default on).

toptenwin

Put the ending display in a SLASH'EM window instead of on stdout (default off). Setting this option makes the score list visible when a windowing version of SLASH'EM is started without a parent window, but it no longer leaves the score list around after game end on a terminal or emulating window.

traps Set the graphics symbols for displaying traps (default "~~~~~#~~~~"). The **traps** option should be followed by a string of 1-22 characters to be used instead of the default traps characters. This string is subjected

to the same processing as the **dungeon** option.

The order of the symbols is: arrow trap, dart trap, falling rock trap, squeaky board, bear trap, land mine, rolling boulder trap, sleeping gas trap, rust trap, fire trap, pit, spiked pit, hole, trap door, teleportation trap, level teleporter, magic portal, web, statue trap, magic trap, anti-magic field, polymorph trap.

Cannot be set with the 'O' command.

verbose

Provide more commentary during the game (default on).

videocolors

Set the color palette for PC systems using NO_TERMS (default 4-2-6-1-5-3-15-12-10-14-9-13-11). The order of colors is red, green, brown, blue, magenta, cyan, bright.white, bright.red, bright.green, yellow, bright.blue, bright.magenta, and bright.cyan. Cannot be set with the 'O' command.

videoshades

Set the intensity level of the three gray scales available (default dark normal light, PC SLASH'EM only). If the game display is difficult to read, try adjusting these scales; if this does not correct the problem, try !color. Cannot be set with the 'O' command.

whitedragonname

Name your starting white dragon (ex. "whitedragonname:Snough"). Cannot be set with the 'O' command.

window type

Select which windowing system to use, such as "tty" or "X11" (default depends on version). Cannot be set with the 'O' command.

wolfname

Name your starting wolf (ex. "wolfname:Beast"). Cannot be set with the 'O' command.

9. Scoring

SLASH'EM maintains a list of the top scores or scorers on your machine, depending on how it is set up. In the latter case, each account on the machine can post only one non-winning score on this list. If you score higher than someone else on this list, or better your previous score, you will be inserted in the proper place under your current name. How many scores are kept can also be set up when SLASH'EM is compiled.

Your score is chiefly based upon how much experience you gained, how much loot you accumulated, how deep you explored, and how the game ended. If you quit the game, you escape with all of your gold intact. If, however, you get killed in the Mazes of Menace, the guild will only hear about 90% of your gold when your corpse is discovered (adventurers have been known to collect finder's fees). So, consider whether you want to take one last hit at that monster and possibly live, or quit and stop with whatever you have. If you quit, you keep all your gold, but if you swing and live, you might find more.

If you just want to see what the current top players/games list is, you can type **SLASH'EM -s all** on most versions.

10. Explore mode

SLASH'EM is an intricate and difficult game. Novices might falter in fear, aware of their ignorance of the means to survive. Well, fear not. Your dungeon may come equipped with an "explore" or "discovery" mode that enables you to keep old save files and cheat death, at the paltry cost of not getting on the high score list.

There are two ways of enabling explore mode. One is to start the game with the **-X** switch. The other is to issue the 'X' command while already playing the game. The other benefits of explore mode are left for the intrepid reader to discover.

11. Credits

The original *hack* game was modeled on the Berkeley UNIX *rogue* game. Large portions of this paper were shamelessly cribbed from *A Guide to the Dungeons of Doom*, by Michael C. Toy and Kenneth C. R. C. Arnold. Small portions were adapted from *Further Exploration of the Dungeons of Doom*, by Ken Arromdee.

NetHack is the product of literally dozens of people's work. Main events in the course of the game development are described below:

Jay Fenlason wrote the original Hack, with help from **Kenny Woodland**, **Mike Thome** and **Jon Payne**.

Andries Brouwer did a major re-write, transforming Hack into a very different game, and published (at least) three versions (1.0.1, 1.0.2, and 1.0.3) for UNIX machines to the Usenet.

Don G. Kneller ported Hack 1.0.3 to Microsoft C and MS-DOS, producing PC HACK 1.01e, added support for DEC Rainbow graphics in version 1.03g, and went on to produce at least four more versions (3.0, 3.2, 3.51, and 3.6).

R. Black ported PC HACK 3.51 to Lattice C and the Atari 520/1040ST, producing ST Hack 1.03.

Mike Stephenson merged these various versions back together, incorporating many of the added features, and produced NetHack 1.4. He then coordinated a cast of thousands in enhancing and debugging NetHack 1.4 and released NetHack versions 2.2 and 2.3.

Later, Mike coordinated a major rewrite of the game, heading a team which included **Ken Arromdee**, **Jean-Christophe Collet**, **Steve Creps**, **Eric Hendrickson**, **Izchak Miller**, **John Rupley**, **Mike Threepoint**, and **Janet Walz**, to produce NetHack 3.0c.

NetHack 3.0 was ported to the Atari by **Eric R. Smith**, to OS/2 by **Timo Hakulinen**, and to VMS by **David Gentzel**. The three of them and **Kevin Darcy** later joined the main development team to produce subsequent revisions of 3.0.

Olaf Seibert ported NetHack 2.3 and 3.0 to the Amiga. **Norm Meluch**, **Stephen Spackman** and **Pierre Martineau** designed overlay code for PC NetHack 3.0. **Johnny Lee** ported NetHack 3.0 to the Macintosh. Along with various other Dungeoneers, they continued to enhance the PC, Macintosh, and Amiga ports through the later revisions of 3.0.

A scant one month before the next major version release of Nethack, two adventurous souls undertook their own modification to the sacred Nethack formula. **Tom Proudfoot** and **Yuval** released Nethack++, which was rapidly renamed Nethack--, contained new monsters, items and other miscellaneous modifications.

Headed by **Mike Stephenson** and coordinated by **Izchak Miller** and **Janet Walz**, the development team which now included **Ken Arromdee**, **David Cohrs**, **Jean-Christophe Collet**, **Kevin Darcy**, **Matt Day**, **Timo Hakulinen**, **Steve Linhart**, **Dean Luick**, **Pat Rankin**, **Eric Raymond**, and **Eric Smith** undertook a radical revision of 3.0. They re-structured the game's design, and re-wrote major parts of the code. They added multiple dungeons, a new display, special individual character quests, a new endgame and many other new features, and produced NetHack 3.1.

Ken Lorber, **Gregg Wonderly** and **Greg Olson**, with help from **Richard Addison**, **Mike Passaretti**, and **Olaf Seibert**, developed NetHack 3.1 for the Amiga.

Norm Meluch and **Kevin Smolkowski**, with help from **Carl Schelin**, **Stephen Spackman**, **Steve VanDeventer**, and **Paul Winner**, ported NetHack 3.1 to the PC.

Jon Watte and **Hao-yang Wang**, with help from **Ross Brown**, **Mike Engber**, **David Hairston**, **Michael Hamel**, **Jonathan Handler**, **Johnny Lee**, **Tim Lennan**, **Rob Menke**, and **Andy Swanson**, developed NetHack 3.1 for the Macintosh, porting it for MPW. Building on their development, **Barton House** added a Think C port.

Timo Hakulinen ported NetHack 3.1 to OS/2. **Eric Smith** ported NetHack 3.1 to the Atari. **Pat Rankin**, with help from **Joshua Delahunty**, is responsible for the VMS version of NetHack 3.1. **Michael Allison** ported NetHack 3.1 to Windows NT.

Dean Luick, with help from **David Cohrs**, developed NetHack 3.1 for X11.

Time passed, and Nethack-- was ported to 3.11 by **Chris**.

Stephen White then released his own modification known as Nethack Plus, which contained new character classes. Unbeknownst to the world at large, **Tom Proudfoot** took this source and combined it with his Nethack--. **Stephen White** went on to add weapon skills, which were eventually integrated into the next version of Nethack, and other features.

In February 1996, **Tom Proudfoot** released SLASH V1. Including part of **Stephen White**'s Nethack Plus and his own Nethack--, leaving unmentioned his own slew of further modifications, this is perhaps the best known of the Nethack modifications. Six versions of this, ending with SLASH V6, are known to exist.

The 3.2 development team, comprised of: **Michael Allison; Ken Arromdee; David Cohrs; Jessie Collet; Steve Creps; Kevin Darcy; Timo Hakulinen; Steve Linhart; Dean Luick; Pat Rankin; Eric Smith; Mike Stephenson; Janet Walz;** and, **Paul Winner**, release version 3.2 in April of 1996.

Version 3.2 marks the tenth anniversary of the formation of the development team. In a testament to their dedication to the game, all thirteen members of the original development team remained on the team at the start of work on the current release. During the interval between the release of 3.1.3 and 3.2, one of the founding members of the development team, **Dr. Izchak Miller**, was diagnosed with cancer and passed away. This release of the game is dedicated to him by the development and porting teams.

Pat Rankin maintained 3.2 for VMS.

Michael Allison, Yitzhak Sapir, and Paul Winner, with help from **Steve Linhart, Kevin Smolkowski, Mike Stephenson, and Stephen White** ported 3.2 for MSDOS.

Keizo Yamamoto and Ken Washikita ported 3.2 for the NEC 98xx machines popular in Japan.

Ken Lorber, Andy Church, and Gregg Wonderly, with help from **Richard Addison**, ported 3.2 for the Amiga.

Dean Luick ported 3.2 to the Macintosh.

Eric Smith and Warwick Allison ported 3.2 for the Ataris.

Michael Allison ported 3.2 for the Microsoft Windows NT platform.

Timo Hakulinen remains responsible for the OS/2 port.

Larry Stewart-Zerba set along a different track - to enhance the spellcasting abilities of the Wizard. Thus, in April 1996, he released version 0.1 of the Wizard Patch. By July, he was joined by **Warwick Allison** and version 0.4 of the Wizard Patch was released. The final update came in September 1996, with the release of Wizard Patch 0.7.

SLASH V6 was picked up by **Enrico Horn** managed to synchronize it with the 3.2 source. The new SLASH 4.1.2 was released as far back as November 1996 went through at least 4 editlevels (E5, E6, E7) with the latest version being 4.1.2E8, synchronized with Nethack 3.2.2 and the Blackmarket option available, released in June 1997.

Nathan La began the arduous task of drawing tiles for the SLASH monsters.

Kentaro Shirakata ported SLASH 4.1.2E8 to Unix.

Lief Clennon ported SLASH 4.1.2E8 to OS/2 EMX.

Romain Dolbeau ported SLASH 4.1.2E8 to Macintosh.

Warren Cheung combined SLASH 4.1.2 and Wizard Patch to create SLASH'EM 0.1 in November 1997. Several revisions including new spells and other minor additions have led to SLASH'EM 0.0.3. **Steven Uy** has generously made additional modifications.

Dirk Schoenberger has continued updating the SLASH/SLASH'EM monster tiles. He has also ported SLASH'EM to Linux.

Lief Clennon ported SLASH'EM to OS/2 EMX.

Kevin Hugo ported SLASH'EM to Macintosh, and has also contributed additional changes and improvements.

Robin Johnson finished arduous task of drawing tiles for the SLASH'EM monsters. He has also contributed many more new tiles.

From time to time, some depraved individual out there in netland sends a particularly intriguing modification to help out with the game. The Gods of the Dungeon sometimes make note of the names of the worst of these miscreants in this, the list of Dungeoneers:

Andy Church	Helge Hafting	Mike Gallop
Andy Swanson	Izchak Miller	Mike Passaretti
Ari Huttunen	Janet Walz	Mike Stephenson
Barton House	Jean-Christophe Collet	Norm Meluch
Benson I. Margulies	Jochen Erwied	Olaf Seibert
Bill Dyer	John Kallen	Pat Rankin
Boudewijn Wayers	John Rupley	Paul Winner
Bruce Holloway	John S. Bien	Pierre Martineau
Bruce Mewborne	Johnny Lee	Ralf Brown
Carl Schelin	Jon W{tte	Richard Addison
David Cohrs	Jonathan Handler	Richard P. Hughey
David Gentzel	Joshua Delahunty	Rob Menke
David Hairston	Keizo Yamamoto	Roland McGrath
Dean Luick	Ken Arromdee	Ross Brown
Del Lamb	Ken Lorber	Scott R. Turner
Deron Meranda	Ken Washikita	Stephen Spackman
Eric Backus	Kevin Darcy	Stephen White
Eric Hendrickson	Kevin Sitze	Steve Creps
Eric R. Smith	Kevin Smolkowski	Steve Linhart
Eric S. Raymond	Kevin Sweet	Steve VanDevender
Frederick Roeber	Mark Gooderum	Tim Lennan
Gil Neiger	Matthew Day	Timo Hakulinen
Greg Laskin	Merlyn LeRoy	Tom Almy
Greg Olson	Michael Allison	Tom West
Gregg Wonderly	Michael Hamel	Warwick Allison
Hao-yang Wang	Michael Sokolov	Yitzhak Sapir

Brand and product names are trademarks or registered trademarks of their respective holders.