

MASSACHUSETTS INSTITUTE OF TECHNOLOGY  
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Working Paper No. 298

February, 1988

**The New Idiot's Guide to OZ**

Liz A. Highleyman

**ABSTRACT:** This is a manual for complete beginners. It assumes no knowledge of the MIT computer systems. This guide will teach you how to log onto the computer called OZ, a DEC PDP-20 computer running the TWENEX (TOPS-20) operating system. You will learn how to use various operating system features, send and receive electronic mail, create and edit files using EMACS, process text using YTeX, and print out your files. This manual has a companion on-line directory on OZ, called <LIZ.GUIDE>, which contains sample programs and examples to use in conjunction with this guide.

© Massachusetts Institute of Technology (1988)

A.I. Laboratory Working Papers are produced for internal circulation, and may contain material that is, for example, too preliminary, too detailed, or too site-specific for formal publication. It is not intended that they should be considered suitable for reference in scientific papers.

## CONTENTS

PREFACE . . . . .	4
ACKNOWLEDGEMENTS . . . . .	4
<b>1. GETTING STARTED . . . . .</b>	<b>5</b>
1.1 Introduction . . . . .	5
1.2 Getting an Account on OZ . . . . .	5
1.3 Note on Terminals and Keyboards . . . . .	5
1.4 Accessing OZ: the Concentrator . . . . .	6
1.5 Logging on to OZ . . . . .	6
1.6 Some Special Features and Operations . . . . .	7
a. Fail-Safe Key Commands . . . . .	7
b. On-Line Help . . . . .	8
c. Using WATSON . . . . .	8
d. The WHOIS and FINGER Commands . . . . .	9
1.7 Detaching and Attaching . . . . .	9
1.8 Killing Jobs and Logging Out . . . . .	9
1.9 A Practice Session . . . . .	10
<b>2. MANIPULATING FILES . . . . .</b>	<b>13</b>
2.1 The Directory . . . . .	13
2.2 Creating, Editing, Saving, and Exiting Files . . . . .	14
2.3 Deleting and Recovering Files . . . . .	15
2.4 Copying Files . . . . .	16
2.5 Restoring Files . . . . .	16
2.6 Using Subdirectories . . . . .	16
2.7 A Practice Session . . . . .	17
<b>3. USING ELECTRONIC MAIL . . . . .</b>	<b>22</b>
3.1 Introduction . . . . .	22
3.2 Using MM . . . . .	22
a. The Basics . . . . .	22
b. Sending Messages . . . . .	22
c. Reading Your Mail . . . . .	23
3.3 Using BABYL . . . . .	24
a. The Basics . . . . .	24
b. Sending Mail with BABYL . . . . .	25
c. Using BABYL to Read Your Mail . . . . .	26

d.	Getting On-Line Help and Exiting <b>BABYL</b> . . . . .	.27
e.	Using <b>BABYL</b> from within <b>EMACS</b> . . . . .	.27
3.4	Maintaining Your Mail Files. . . . .	.27
3.5	Mail Short Cuts. . . . .	.28
3.6	Using Mailing Lists . . . . .	.28
3.7	Long Distance: Using Networks . . . . .	.29
3.8	A Practice Session. . . . .	.29
4.	<b>EDITING WITH EMACS</b> . . . . .	.33
4.1	Introduction . . . . .	.33
4.2	On-Line Help and the <b>TEACH-EMACS</b> Tutorial. . . . .	.33
4.3	Using Buffers. . . . .	.33
4.4	Entering <b>EMACS</b> to Edit or Create a File . . . . .	.34
4.5	Using Windows Within <b>EMACS</b> . . . . .	.34
4.6	Basic Editing Commands . . . . .	.35
a.	The Status Line . . . . .	.35
b.	Moving Within the File. . . . .	.35
c.	Inserting and Transposing Text . . . . .	.36
d.	Deleting and Undeleting Text . . . . .	.36
4.7	Searching for Strings of Text. . . . .	.36
4.8	Moving and Copying Blocks of Text . . . . .	.37
4.9	Some Special Character Commands . . . . .	.37
4.10	Extended ( <b>M-X</b> ) Commands . . . . .	.38
4.11	Using Keyboard Macros. . . . .	.40
4.12	Saving the File and Exiting <b>EMACS</b> . . . . .	.40
4.13	A Practice Session. . . . .	.40
5.	<b>TEXT FORMATTING WITH YTeX</b> . . . . .	.43
5.1	Introduction . . . . .	.43
5.2	Using <b>YTeX</b> : The Basic Idea . . . . .	.43
5.3	<b>YTeX</b> Opening Commands . . . . .	.44
a.	Setting Sizes. . . . .	.44
b.	Inputting Auxiliary Files. . . . .	.44
c.	Defining Macros . . . . .	.44
5.4	Stylistic Commands. . . . .	.45
a.	Layout Commands . . . . .	.45
b.	Spacing Commands. . . . .	.47
c.	Font Changing Commands . . . . .	.47
d.	Special Characters. . . . .	.48
5.5	Formatting Equations . . . . .	.48

a. Horizontal Mode. . . . .	.48
b. Vertical Mode. . . . .	.49
c. Some Basic Equation Commands . . . . .	.50
5.6 Creating Tables. . . . .	.50
5.7 Closing Commands . . . . .	.51
5.8 Running the Y $\TeX$ Program and Debugging the Input File. . . . .	.51
a. Running Y $\TeX$ . . . . .	.51
b. The .DVI and .LST Files . . . . .	.52
c. Debugging the .Y $\TeX$ File. . . . .	.52
5.9 A Practice Session. . . . .	.53
6. PRINTING FILES. . . . .	.59
6.1 Printers Available at the A.I. Laboratory . . . . .	.59
6.2 Checking the Printer Status . . . . .	.59
6.3 Printing a File. . . . .	.60
a. Defining a Printer. . . . .	.60
b. Sending a Job to the Printer . . . . .	.60
c. Stopping a Print Job. . . . .	.60
6.4 Printing Errors . . . . .	.61
6.5 A Practice Session. . . . .	.61
7. APPENDICES . . . . .	.64
7.A Command Summary . . . . .	.64
7.B Glossary of Terms. . . . .	.75
7.C Other Resources . . . . .	.81

## PREFACE

This is a manual for complete beginners. You will learn how to log onto the computer called OZ, a DEC PDP-20 computer running the TWENEX (TOPS-20) operating system. You will learn how to use various operating system features, send and receive electronic mail, create and edit files using the EMACS editor, process text using YTeX, and print out your files. The EMACS editor and YTeX text formatting program will allow you to do advanced word processing and achieve professional looking results. YTeX allows for the complex setting of mathematical equations that is often necessary in scientific and technical papers.

This manual is organized so that topics are presented in the likely order in which you will use them. For the sake of coherence, all material regarding a given topic has been kept together.

There are certain conventions you should keep in mind as you use this manual. The *control key* is represented by the symbol ↑. The *meta key* will be either the <pause> key or the <escape> key, depending on your terminal. This key will be represented by an M-. A carriage return is represented as <CR>. Computer commands, program names, file names, and computer output will be shown in typewriter font. Commands will be shown in CAPITAL LETTERS. OZ generally responds in the same way to both upper- and lower-case letters; this convention is adopted to improve readability. [Square brackets] indicate that the words within should be replaced by something specific. For example [USERNAME] should be replaced by the appropriate username, omitting the brackets. *Italics* indicate new words, which are defined in the glossary at the end of this guide.

At the end of each section you will find step-by-step instructions for going through a practice session using the files provided in the directory <LIZ.GUIDE>. Each practice session will cover the new material presented in that section. The lines flush with the margin are what will appear on your screen. The lines indented and marked by an arrow (==>) indicate responses to be entered by the user.

As a resource for beginners, this manual is necessarily incomplete. At the end of the guide you will find a list of more advanced manuals and sources that you may wish to use once you have mastered the material in the *New Idiot's Guide*. Any errors found in this guide or suggestions for improvement should be reported to LIZ@OZ.

## ACKNOWLEDGEMENTS

The original *Idiot's Guide to OZ* was written in 1984 by William Gilson, under the aegis of Whitman Richard's Natural Computation Group. The work on the original guide was supported by grants provided by the NSF and AFOSR. Version 1 of *The New Idiot's Guide to OZ* was written updated at the Artificial Intelligence Laboratory in August 1987. The guide was updated to Version 2 in February 1988. Many thanks to those who offered advice and comments, especially Claudia Smith for all her assistance.

## 1. GETTING STARTED

### 1.1 Introduction

OZ is a DEC PDP-20 computer located on the ninth floor of the A.I. Laboratory. The *operating system* on OZ, known as TWENEX, or TOPS-20, is the top level program that enables multiple users to work easily on the computer.

### 1.2 Getting an Account on OZ

Before you begin, you will need to get an *account* on OZ. This will give you a *username*, also known as a *login name*. You will choose your username when your account is set up. The username is most commonly the user's initials or nickname. You may also choose a password, and a certain amount of memory space will be allocated for your use. Ask your supervisor or sponsor how to go about getting an account.

### 1.3 Note on Terminals and Keyboards

There are a number of different types of terminals in use at the A.I. Laboratory. You may be using one of the Ann Arbor terminals or a LISP Machine. The keyboard layout on these terminals varies quite a bit. Associated with your personal directory should be a file called LOGIN.CMD. This will contain commands to set the parameters specific to your terminal. After you have learned about files (see Section 2), you can look at and copy the file <LIZ.GUIDE>LOGIN.CMD into your own directory, or you can ask the person who sets up your account to initiate your LOGIN.CMD file. When you log in, your LOGIN.CMD file will automatically be read and executed. If you wish to re-initialize your environment while logged in, type TAKE LOGIN.CMD <CR> to re-execute your LOGIN.CMD file.

There are certain special keys that you should be familiar with. The <delete> key, labelled <del> or <rubout> on some keyboards, will delete the text preceding the cursor. The <back space> key backs up the cursor without deleting any material.

Many commands require the use of the *control key*. This is to the left of the letter keys on most terminals. In this manual, the control key is represented by the symbol ↑. ↑C indicates that you should press the control key, and **simultaneously** press the desired letter key, in this case C.

Other special keys are the *meta*, *alt*, and *escape* keys. Use of these keys will vary with keyboards. The *meta key* is used to give different meanings to letter commands. The Ann Arbor Ambassador terminals use the <pause> key, located in the lower left-hand corner, as its meta key. The Ann Arbor XL keyboard uses the <escape> key, at the upper left, as its meta. In this guide, the meta key will be represented by the symbol M-. For example, M-v

indicates that you should either press the <pause> key or the <escape> key, and then the desired letter, in this case v. If you are using an XL terminal, it will be necessary in some cases to press both the <escape> and <shift> keys. Do this if the letter following the M is shown in upper-case (M-X).

#### 1.4 Accessing OZ: the Concentrator

You can access OZ from your terminal via the *concentrator*, a router that connects the different terminals to the various computers at the lab.

When you first approach your terminal, you will see a message similar to this:

```
AI LAB NE43-7A-HUB MINUTES - Type ? for help.>
```

To access OZ, type 14|Z after the >. This tells the the concentrator which computer you wish to use.

If the computer responds with the message:

Trying...

it is likely that all the access lines are currently in use. The best thing to do is to wait a few minutes and try again. To get back to the opening *prompt*, type ↑\k. If the system does not respond after a number of tries, the computer may be *down*, and you will have to postpone your session. Try to access OZ periodically, since most outages are of short duration and the computer will not give you a signal when it comes back *up*.

It is also possible to access OZ using the TELNET or SUPDUP protocols. At the opening > prompt, type ? to see the available computers and protocol options. Typing OT or 14T will access OZ via the TELNET protocol. When using this protocol, you will be asked for your password in order to complete the login process. If you are using a personal computer to access OZ, it may be necessary to use 14S, the SUPDUP protocol. If you are accessing OZ from a LISP Machine, type <select> <help> to list available options, and enter T for terminal mode.

You can also use the concentrator to access the weather computer. Type OW or 14W at the opening > prompt. The weather forecast will be displayed on your screen, and you will be returned to the original prompt.

#### 1.5 Logging on to OZ

Once you have accessed OZ, your screen *prompt* will be @ (called the *at-sign*). The @ indicates that you are at the *exec*, or *top level* of OZ. Exec is the level at which you execute *jobs*, or *programs*. To start your session, type [USERNAME] <CR> if you have logged in using

14↑Z, or [USERNAME] [PASSWORD] <CR> if you have used the TELNET protocol (14T). Once you have logged on, you will see a notice on your screen saying how long the system has been up and how many jobs are currently in progress. Your screen may also say --Msgs--. To see the system messages, type <CR>. If the messages do not all fit on the screen, you will see --Pause-- at the bottom. Hit the <space bar> to see the remainder. If you do not wish to see these messages, type N <CR> (for "No more"). After the messages have been displayed or skipped, you will be returned to the @ prompt. You are now logged on to OZ.

Because of the academic environment at the A.I. Laboratory and the non-confidential nature of most work, computer security at the lab is not strict. This policy assumes of course, that users will not abuse this privilege by prying into or altering other users' files. You may wish to periodically change your password as a security measure. To do this, type PASSWD <CR> at the @ prompt. OZ will ask you to type in your old (current) password, and the new one twice.

## 1.6 Some Special Features and Operations

### a. Fail-Safe Key Commands

If you ever type a command that OZ does not understand, it will reply: ?Eh. When this occurs, try re-entering your command, being careful to type it correctly.

For most of the commands you will use, there is a two or three letter abbreviation that can be used instead. For example, DEL can be used in place of DELETE. OZ also has a feature that will complete your commands for you. Type enough letters of the command so that it cannot be confused with any other, and press <escape>. The full command will appear on the screen. If it is correct, hit <CR> and the command will be executed. If not, use the <delete> or <rubout> key to amend it.

If at any time you want to exit whatever program you are working with and return to the exec level, type ↑C. This will return you to the top level and you will see the @ prompt. It is usually a good policy to try to use ↑C if you get stuck in a program. Try it a second time if you get no response.

If you wish to clear your video screen, type ↑L. This may be necessary if, for example, you receive mail in the middle of an editing session, or if odd messages appear. If you are at the top level, the screen will be cleared and the @ prompt will appear at the top. If you are in the editor when this command is given, the text you are working on will be redrawn without the extraneous material.

If you have typed a command that you decide you do not wish to execute, type ↑U to erase the command and return to the @.



If you get stuck, and if ↑C does not work and the screen remains frozen, this may be the sign of a system *crash*. The best thing to do is type ↑\k to return to the original > prompt and try logging in again later.

## b. On-Line Help

In learning to use OZ, you should know that there is much helpful information *on-line*, that is, within the computer itself. In general it is a good idea, when confronted with a problem, to try to find the answer within the system, as this will save you the time and effort of finding the appropriate manuals. It can also lead to some interesting discoveries and helpful hints about using OZ.

At top level, type `HELP <CR>`. On the screen will be listed three possible starting points: 1) `HELP * <CR>`; 2) `HELP ? <CR>`; and 3) `HELP [TOPIC NAME] <CR>`. The first will give you a listing of all topics for which on-line *documentation* is available. The second command will give an abbreviated list, containing the more commonly requested topics. Once you have looked at these lists, select the topic about which you would like more information and use the third command, `HELP [TOPIC NAME] <CR>`, where [TOPIC NAME] is replaced with the desired subject from the lists. The command `HELP DOCUMENTATION <CR>` will list some of the on-line documentation available.

Most programs, such as EMACS and the mail programs MM and BABYL, have their own built-in documentation. Often typing `HELP <CR>` or `?<CR>` while within a program will show you what kind of on-line documentation is available and how to get it.

## c. Using WATSON

When you first log in as a new user, you should run the WATSON program (also known as INQUIRE) and answer the questions it asks you. This will provide OZ with some basic information about yourself which will be useful to other users. To run this program, type `WATSON <CR>` at the @ prompt. The program is easy to use, and it provides fairly complete instructions. Start by entering your username, then follow the directions as you go along. The program will ask for information such as your MIT address and phone extension, birthday, supervisor, and projects you are working on. The Remarks entry allows you to enter your own personal comment or witticism. If you get stuck, enter `?<CR>`. This will give you a list of the different possible entries. If you enter `ALL <CR>`, you will be asked to supply information for all the entries (although you may skip those you do not wish to answer). When finished, type `DONE <CR>`, and WATSON will update your entry.

#### d. The WHOIS and FINGER Commands

To see the results of what you have typed into WATSON, type `WHOIS [YOUR USERNAME] <CR>` at the `@` prompt. You may use the `WHOIS` command with any person's username, and also with a last name if the username is unknown.

If you type `WHOIS <CR>`, without a specific name, you will get a listing of all the users logged on to OZ, their WATSON entries, and what programs they are currently running.

A simpler way to see who is logged in to OZ is to type `FINGER <CR>`. A list will appear on the screen giving each person's username, their real name, the program they are running, and the location of their terminal. You can also type `FINGER [USERNAME] <CR>` to get the same information about a specific user. The command `SYS <CR>` will give you similar information, but in a more abbreviated form.

#### 1.7 Detaching and Attaching

When you leave your terminal idle for any length of time you may wish to detach your jobs so that they will not be interfered with. To do this, type `DETACH <CR>` at the `@` prompt. This will return the terminal to the opening `>` prompt and allow other users to log in to the terminal. Upon returning, type `14↑Z` or `14T` to access OZ and log on as you normally would. When you do this, you will see the message on your screen: `Attach your detached job? [Confirm]`. Hit `<CR>` to re-attach. When you do this, the job you were previously working on, and all features of your environment (except those affected by your `LOGIN.CMD` file), will be returned to the state they were in when you detached. It may also be necessary to re-attach your job after a system crash; do so if the prompt shown above appears on your screen. If the prompt does not appear, you can re-attach your job by typing `ATTACH [USERNAME] <CR>` at the top level.

#### 1.8 Killing Jobs and Logging Out

When you have finished your session on OZ, you will need to *kill* any jobs that are currently in progress and log out. In order to see what jobs, or *forks*, are currently running, type the command `INF FO <CR>` ("INformation about FOrk status"). The screen will display a list of your existing jobs and their current status. When you return to the exec level from a program, the program is not automatically killed. To kill a job, use the `RESET` command. For example, to kill an `EMACS` job, type `RESET EMACS <CR>`. To kill the most recent fork used, type `RESET . <CR>`. To kill all your current jobs at once, type `RESET * <CR>`. The `*` is a *wildcard* symbol, used to represent all possibilities.

When you have killed your jobs and are ready to leave the system, type `LOGOUT <CR>` or `KK <CR>`.

## 1.9 A Practice Session

This practice session will start you off by showing you how to log in and use some of the special features of OZ. In this and the following Practice Sessions, the lines flush with the margin are what will appear on your screen. The lines indented and marked by arrows (==>) indicate appropriate user responses.

```
AI LAB NE43-7A-HUB MINUTES - Type ? for help.>
```

```
==> 14↑Z
```

```
Trying...
```

```
MIT-OZ Artificial Intelligence Laboratory, TOPS-20 Monitor 5(6032)-1
```

```
The system has been up for 13 hours and 33 minutes
```

```
There are 50+5 jobs and the load average is 3.45
```

```
@
```

```
==> [LOGIN NAME] <CR>
```

```
Job 37 on TTY116 1-Jul-87 15:35
```

```
Last logout: 30-Jun-87 17:30 from CHAOSnet host NE43-7A-HUB.AI.MIT.EDU
```

```
Wouldn't opera be wonderful if there were no singers?
```

```
End of LOGIN.CMD.5
```

```
You have mail from POGGIO at 13:02
```

```
--Msgs--
```

```
==> <CR> or <space bar> (to see the system messages)
```

If the message is too long to fit on your screen, you will see:

```
--Pause--
```

at the bottom of the screen. Press <space bar> to see the remainder.

```
@
```

```
==> WATSON <CR>
```

Type your username. If you are intending to change your username, type the old one here. To select someone else, type her username. If you don't know how to use WATSON, type ?<RETURN>.

Is this right for Username?

Old: [USERNAME]

If so, hit <RETURN>. If not, type the correct entry and a <RETURN>. Type <ESCAPE> or <ALT> to clear the entry.

New:

==> <CR>

The USERNAME entry is for "Joe J. Username"

If this isn't you, type <RETURN>. The LIST command prints all the information on you specified so far.

What next? (Type "?" for a list of commands, "HELP" for help with this program, or "DONE" to exit and make these changes permanent.)

WATSON>

==> ALL <CR>

NOTE: Here WATSON will run through all the categories. Supply the information requested, using <CR> to bypass any entries that you wish to leave blank or do not understand. After you have gone through all the entries, WATSON will display the information that has been provided, and return to the prompt:

WATSON>

==> DONE <CR>

Sending update request to Daemon...OK

Thank you, and goodbye from Watson

@

@

==> WHOIS [YOUR USERNAME] <CR>

```

USERNAME  A-Joe J. Username          EMACS .116 NE43-7A-HUB (Chaos)
          (JOE) [username@OZ.AI.MIT.EDU] Hacking Vision for T. Poggio
          Birthday April 7; Work NE43-772; 3-6666
          Home 84 Rat Terrace, Boston; 555-1212

```

Time flies like the wind...

...Fruit flies like bananas

⓪

==> FINGER <CR>

-User-	GR --Full Name--	-What-	Idle	TTY	-Console location-
ANITA	Anita M. Flynn	EMACS	1.15	142	HUREWICZ (Chaos)
BVHS	Bror V.H. Saxberg	EXEC	1.08	151	SNOOPY (Chaos)
LIZ	Liz A. Highleyman	EXEC	.33	116	NE43-7A-HUB (Chaos)
MCK	M. Cathleen Krebs	EMACS	1.22	6	NE43-8B-HUB (Chaos)
TOMR	Thomas J. Reinhardt	MM	1.00	123	APIARY-3

⓪

==> INFO FO <CR>

MM(1): Kept, HALT at 116251, 0:00:01.9

EMACS(2): Editor, Kept, HALT at 52047, 0:00:04.9

NOTE: These will vary depending on which programs you have most recently used.

⓪

==> RESET \* <CR>

⓪

==> LOGOUT <CR>

Logout Job 13, User [USERNAME], Account, TTY 116,  
at 1-Jul-87 18:30.55, Used 0:01:03 in 0:51:00

[Host connection closed.]

## 2. MANIPULATING FILES

### 2.1 The Directory

As a user with an account on OZ, you will have a *directory* associated with your username. Every user has a main, or *home* or *root directory*, which contains his *files*. The name of a user's home directory is usually the same as his username. Users may also have a number of associated *subdirectories*.

In order to see the contents of your directory, type DIRECTORY <CR>, or DIR <CR>. You will see a listing similar to the following:

```
OZ:<[USERNAME]>
EXAMPLE.TXT.1
IDIOTS-GUIDE.YTEX.61,62,63
LOGIN.CMD.1
LOGOUT.CMD.1
MAIL.TXT.1
[USERNAME].BABYL.1
WELCOME.TXT.1

Total of 8 files
```

Most of the files in your own home directory will have different names than the ones listed above. The directory listing will change every time you store or remove something from OZ's memory.

Each file name has three parts, separated by dots. The first part gives an indication of the contents of the file. The second part usually indicates the type of file. For example .TXT files contain text, .CMD files contain commands, and .LSP files contain LISP programs. Files containing commands to be used by one of the text formatting programs should have a suffix such as .YTEX. Formatted files will end with the suffix .DVI. You are free to name your files whatever you like, but following these conventions makes it easier to remember the file type. If you want to use more than one word in the first part of the file name, separate the words by dashes, as in IDIOTS-GUIDE.YTEX. The third part of the file name is the version number. Each time you change one of your files, OZ will create a new *version*. OZ will usually keep three versions of each file. You do not need to include the version number when you name a file; OZ will append this automatically. When editing or visiting a file, if you do not specify a version number, OZ will assume that you want the latest version.

If you wish to see a more detailed listing of your directory, you can use the command VDI <CR> (for View Directory). Your listing of files will look like the following:

```

OZ:<[USERNAME]>
EXAMPLE.TXT.1;P777777      1 333(7)      19-May-87 13:45:09 [USERNAME]
IDIOTS-GUIDE.YTEX.61;P777777 47 120122(7)  10-Jul-87 15:52:56 [USERNAME]
    .62;P777777          48 120518(7)  10-Jul-87 15:59:06 [USERNAME]
    .63;P777777          48 120566(7)  13-Jul-87 11:50:52 [USERNAME]
LOGIN.CMD.1;P777777        1 188(7)      10-Jul-87 16:42:36 [USERNAME]
LOGOUT.CMD.1;P777777       1 51(7)        2-Jul-87 13:22:55 [USERNAME]
MAIL.TXT.1;P777777        2 321(7)      13-Jul-87 10:25:09 [USERNAME]
    .2;P777777          2 350(7)      14-Jul-87 13:09:43 [USERNAME]
[USERNAME].BABYL.1;P777777  1 207(7)      10-Jun-87 11:52:53 [USERNAME]
WELCOME.TXT.1;P777777     12 3105(7)     7-Apr-87 16:33:36 [USERNAME]

```

Total of 162 pages in 10 files

This listing indicates the names of the files, their size in pages, the date and time they were last modified and saved, and the user who saved them.

An easy way to specify a file is to type enough letters of the file name so that the file can be uniquely identified. Then press the <escape> key and the file name will be completed. You can also specify files using \*, the wildcard, in place of part of the filename. For example, typing DIR LOG\*.\* <CR> would list the files LOGIN.CMD.1 and LOGOUT.CMD.1 because both start with "LOG". This feature is useful if you are not sure of a complete filename.

To see the files in another user's directory, type DIR <[DIRECTORY NAME]> <CR>. The directory name for a user's home directory is usually the same as his username. To connect to another user's directory, type CD <[DIRECTORY NAME]> <CR> ("Change Directory"). Until you change back to your own directory, your work will be saved in the directory you are currently using. To enter another user's directory, type the name of that directory at the @ prompt. When you are in another user's directory, the DIR <CR> command will list his files. To return to your own directory, type your directory name at the @ prompt.

## 2.2 Creating, Editing, Saving, and Exiting Files

There is more than one way to create files on OZ. The first is by *visiting* the file. At the @ prompt, type EMACS <CR> to initiate an EMACS job (assuming that you do not have any other EMACS jobs currently in progress). You will see the following at the top of your screen:

```
EMACS Editor, version 165 - type ^_ (the help character) for help.
```

Now press ↑X|V (for "Visit"). At the bottom of your screen, the following will appear:

```
Visit File(Default OZ:<[USERNAME]>GAZONK.DEL.0):
```

After the colon, type the file name you wish to give the new file. (**CAUTION:** Do not give the new file a name that is already in use. If you do this, you will write over whatever was in the file that originally had that name.) Press <CR>, and the new file will be started. There will be a note at the bottom of the screen which gives the filename and says (**New File**). You will now be in the editor, and can type in whatever you wish (see Section 4 on using the EMACS editor). When you are ready to exit, save the file by typing ↑X↑S. The version number will change from 0 to 1, and the file will be written. Type ↑X↑Z to leave the editor. (Typing ↑Z↑Z will have the same effect.) If you decide not to create the file, just type ↑X↑Z to exit the editor without writing the file.

A simpler way to create a file is to type EMACS [NEW FILENAME] <CR> or EDIT [NEW FILENAME] <CR> at the @ prompt. If any text is typed into the file, ↑X↑S will save it, and ↑X↑Z will take you out of the editor. If no text is entered, ↑X↑Z will exit the editor without creating the new file. If you type EMACS <CR> or EDIT <CR> without specifying a file name, you will be returned to the last file you worked on.

### 2.3 Deleting and Recovering Files

In order to remove a file you no longer need, type DELETE [FILE NAME] <CR>, or DEL [FILE NAME] <CR>. The name of the deleted file(s) will appear on your screen along with the notation: [OK]. You will see when you list your directory using DIR <CR> that the deleted files no longer appear in your listing. They are not really gone from memory, however. In order to remove the files, type EXPUNGE <CR> at the @ prompt. If you have deleted a file by mistake but have not yet used EXPUNGE, the command UNDELETE [FILEMANE] <CR> will bring it back. If you ask OZ to delete a file without specifying a version number, it will delete all versions of that file.

In order to conserve memory space on OZ, it is good policy to clean up your directory periodically to remove old versions and files you no longer need. The command PURGE <CR> will delete all but the latest version of each file. The command PURGE 2 <CR> will delete all but the two most recent versions. It is a good idea to keep at least one backup version of your important files. When you use either command, you will be shown a list of all the files that have been marked for removal, and you will be asked if these files can be expunged. If you do not want to keep any of the files, type Y. If you do wish to keep some of the files, type N, then UNDELETE [FILENAME] <CR> the desired files.

In some cases, it may be possible to recover files that have been expunged. Use the command GROVEL <CR> to enter a program that will help you to locate your files on backup tapes and restore them. Type HELP GROVEL <CR> for more information.



## 2.4 Copying Files

Copying files on OZ is a simple procedure. Type `COPY [CURRENT FILE NAME] [NEW FILE NAME] <CR>`. The current file name is the name of the file you want to copy. The new file name is whatever you wish to name the duplicate file. Again, be sure that the name is one that is not already used in the directory.

When copying files within a single directory, you do not need to specify the directory name. If you are copying from one directory to another, the copy command will be as follows:

```
COPY <[DIRECTORY NAME1]>[FILENAME] <[DIRECTORY NAME2]>[FILENAME] <CR>.
```

## 2.5 Restoring Files

In order to keep OZ's memory from filling up, files that have not been used for a substantial amount of time will be taken *offline*. The `DIR <CR>` command will show you the status of your files. *Migrated*, or archived, files will have the designation `;OFFLINE` following the filename. The command `NDIR <CR>` will show you when and where the migrated files were moved. These files have been backed up onto tapes. To restore off-line files, use the command `RETRIEVE [FILENAME] <CR>`. This adds your desired files to the retrieval queue, which is run a couple of times a week by the tape archivist. When the files are restored, the `;OFFLINE` flag will disappear from your directory listing. Type `HELP RETRIEVE <CR>` for more information.

## 2.6 Using Subdirectories

Users who have a large number of files on the system often have them arranged in *subdirectories*. This is a good way to keep files organized. You will probably not have any need for subdirectories until you have accumulated many files, but you may need to access the subdirectories of other users. A subdirectory name is of the form `<[HOME-DIRECTORY].[SUB-NAME]>`, for example, `<LIZ.GUIDE>`. To see a list of the subdirectories associated with a particular home directory, type `DIR <[HOME-DIRECTORY].*> <CR>`. In a normal `DIR` listing, subdirectories will appear in the listing as `[SUB-NAME].DIRECTORY.1`. To enter a subdirectory, type the subdirectory name (including the home directory prefix) at the `@` prompt.

To create a subdirectory, type the command `BUILD [SUBDIRECTORY NAME] <CR>` at the `@` prompt. Your prompt will then be `@@`. Type `LIST <CR>` to see the current attributes of the subdirectory. Typing `? <CR>` will show you a list of possible commands. It will most likely be necessary to increase your subdirectory capacity and memory (page quotas) in order to create subdirectories. Within your home directory, the command to increase

your subdirectory capacity is MAX <ESC> [NUMBER] <CR>; 100 should be sufficient. Within your home directory or a subdirectory, the commands to increase your page quotas are WORK [NUMBER] <CR> (for the working page quota) and PERM [NUMBER] <CR> (for the permanent quota). Set quotas at 500 to start. If you receive messages telling you that your quota has been exceeded, increase your quotas further. To exit the BUILD program, type <CR> once more. It is also possible to create subdirectories within other subdirectories. For more information, or if you lack the permissions necessary to create subdirectories, consult an expert.

## 2.7 A Practice Session

Assuming you are still logged on to OZ and at the exec level, this Practice Session will show you how to create, save, delete and copy files. The files in your personal home directory will not be the same as those shown in the example listing.

⓪

```
==> DIR <CR>
```

```
OZ:<[USERNAME]>
IDIOTS-GUIDE.YTEX.61,62,63
LOGIN.CMD.1
LOGOUT.CMD.1
MAIL.TXT.1
[USERNAME].BABYL.1
WELCOME.TXT.1

Total of 8 files
```

⓪

```
==> COPY MAIL.TXT <CR>
```

⓪

```
==> DIR <CR>
```

```
OZ:<[USERNAME]>
IDIOTS-GUIDE.YTEX.61,62,63
LOGIN.CMD.1
LOGOUT.CMD.1
MAIL.TXT.1,2
[USERNAME].BABYL.1
```

WELCOME.TXT.1

Total of 9 files

**NOTE: Copying files (a)** - You can see that there are now two versions of the file MAIL.TXT, and that the file total has increased by one.

Ⓢ

==> COPY WELCOME.TXT HELLO.TXT <CR>

OZ:<[USERNAME]>

IDIOTS-GUIDE.YTEX.61,62,63

HELLO.TXT.1

LOGIN.CMD.1

LOGOUT.CMD.1

MAIL.TXT.1,2

[USERNAME].BABYL.1

WELCOME.TXT.1

Total of 10 files

Ⓢ

**NOTE: Copying files (b)** - There is now a new file, named HELLO.TXT.1. This file has the same contents as the file WELCOME.TXT.1, so you can remove the old one.

Ⓢ

==> DELETE WELCOME.TXT <CR>

WELCOME.TXT.1 [OK]

Ⓢ

==> DIR <CR>

OZ:<[USERNAME]>

IDIOTS-GUIDE.YTEX.61,62,63

HELLO.TXT.1

LOGIN.CMD.1

LOGOUT.CMD.1

MAIL.TXT.1,2

[USERNAME].BABYL.1

Total of 9 files

**NOTE: Deleting a file**

⓪

==&gt; EXPUNGE &lt;CR&gt;

OZ:&lt;LIZ.GUIDE&gt; [1 page freed]

**NOTE:** In order to use the file <LIZ.GUIDE>EXAMPLE.TXT, which you will need in Section 4, you will have to copy the file into your directory from the directory <LIZ.GUIDE>. This assures that an original copy of the file is kept in the old directory for other users, and gives you your own copy to use as a practice file:

⓪

==&gt; COPY &lt;LIZ.GUIDE&gt;EXAMPLE.TXT &lt;[YOUR DIRECTORY NAME]&gt;.EXAMPLE.TXT &lt;CR&gt;

⓪

==&gt; EMACS &lt;CR&gt;

EMACS Editor, version 165 - type ^\_ (the help character) for help.

==&gt; ↑X↑V

Visit File (Default OZ:&lt;[USERNAME]&gt;GAZONK.DEL.0):

==&gt; FOO.TXT &lt;CR&gt;

**NOTE: Creating a file** - You have now entered the EMACS editor to create a new file. The following will appear at the bottom of your screen:

EMACS (Text Fill) MAIN: OZ:<[USERNAME]>FOO.TXT (0)  
(New File)

**NOTE:** At this point, enter the text of the file:

```
==> This is a sample file <CR>
    ↑X↑S
    ↑Z↑Z
```

**NOTE: Saving the file** - ↑X↑S writes the file; ↑X↑Z exits the editor. You will see the following at the bottom of your screen:

Written: OZ:<[USERNAME]>FOO.TXT (1)

@

==> COPY FOO.TXT FOO2.TXT <CR>

@

==> DIR <CR>

OZ:<[USERNAME]>

EXAMPLE.TXT.1

FOO.TXT.1

FOO2.TXT.1

IDIOTS-GUIDE.YTEX.61,62,63

HELLO.TXT.1

LOGIN.CMD.1

LOGOUT.CMD.1

MAIL.TXT.1,2

[USERNAME].BABYL.1

Total of 12 files

**NOTE: Deletion using a wildcard** - Since the FOO files are just sample files, you can erase them. Use the wildcard, \*, to delete both at once:

@

==> DEL FOO\*.\* <CR>

FOO.TXT.1 [OK]

FOO2.TXT.1 [OK]

@

==> EXPUNGE <CR>

OZ:<[USERNAME]> [2 pages freed]

@

==> PURGE <CR>

These are the deleted files for directory OZ:<[USERNAME]>:

IDIOTS-GUIDE.YTEX.61      IDIOTS-GUIDE.YTEX.62      MAIL.TXT.1

Total of 3 files

Expunge?

==> Y

[OK -- 70 Pages Freed.]

Total Remaining Pages in Directory: 320

Ⓞ

==> DIR <CR>

OZ:<[USERNAME]>

EXAMPLE.TXT.1

IDIOTS-GUIDE.YTEX.63

HELLO.TXT.1

LOGIN.CMD.1

LOGOUT.CMD.1

MAIL.TXT.2

[USERNAME].BABYL.1

Total of 7 files

**NOTE:** Only the latest version of each file remains.

### 3. USING ELECTRONIC MAIL

#### 3.1 Introduction

There are a variety of ways of receiving and sending mail on OZ. There are two mail programs you will learn in this section, MM and BABYL. MM is easier to use, but BABYL is more sophisticated and has more features. You will also learn how to send mail from the exec level, how to use mailing lists, and how to send long distance mail over the network.

#### 3.2 Using MM

##### a. The Basics

To use the MM mail program, type MM at the top level (at the @ prompt). This will put you into the program and list any messages you have not yet read. Within the MM program, your prompt will be MM>. At this prompt, you can either type SEND <CR>, to enter the sending subprogram, or READ <CR>, to enter the reading subprogram.

Like many of the programs on OZ, MM has a certain amount of helpful on-line information available to the user. Type ?<CR> at the MM> prompt. You will be shown a list of commands available at this level of MM. If you are curious about what any one of them does, type HELP [COMMAND NAME] <CR>. Note that there is a general command available at this level of MM which gives some basic information: type HELP <CR>.

##### b. Sending Messages

At the MM> prompt, type SEND <CR> to enter the sending subprogram (this also will work if you type MAIL <CR>). The computer will respond with the prompt To: Here, enter the username (login name) of the person to whom you wish to send the message. The computer will respond with cc: Enter the username of anyone to whom you wish to send a copy of the message (this can be yourself); if you do not wish to send copies, just press <CR>. (NOTE: This program will not work with a person's real name. You can use the command WHOIS [LAST NAME] <CR> at the top level to find out his username.) The next prompt will be Subject: Type in a short description of the message content. After you have done this, press <CR> and the following will appear on your screen:

Message (End with ESCAPE or CTRL/Z.

Use CTRL/B to insert a file, CTRL/E to enter editor, CRL/K to redisplay message, CTRL/L to clear screen and redisplay, CTRL/N to abort.):

At this point, you can type in your message. It can be as long as you wish; use a <CR> between lines. When you have completed the message, type ↑Z. This will remove you from

the message portion of the program, and put you at the S> prompt. Type SEND <CR>. This will send your message, and you will see a note on the screen telling you that the message has been *queued*, that is, put in the recipient's mail receiving file, and will be seen the next time she reads her mail.

If you wish to change your message, you can use ↑E to enter the editor. The editing commands are the same as those for EMACS, which you will learn about in Section 4.

If you decide not to send the message, type ↑N to kill it. The computer will respond with: Abort?. Type Y, and you will be returned to the MM> prompt.

Once you are proficient at manipulating files (Section 2), you may find the ↑B command useful. Create a file containing your message; it is helpful to name this file FILENAME.SEND or FILENAME.MSG for mnemonic purposes. Enter the MM program, type SEND <CR> and enter the heading information as described above. When you come to the message portion, type ↑B. The computer will respond (Insert file: Type the filename. Once the file has been read in completely, your screen will say EOF (for End Of File). At this point, you can enter any additional information you want to add, or just type ↑Z to end the message. Type SEND <CR> at the S> prompt. This feature is useful if you want to send the same message repeatedly without retyping it. You can also use ↑B to transfer text files to other users.

### c. Reading your Mail

Normally, when you enter the MM program, something like this will appear:

```
File: <[USERNAME]>MAIL.TXT.1
There are 2 additional messages
N      1) 1-Jul To: username@OZ.AI.M Paper wanted (235 chars)
N      2) 1-Jul To: username@OZ.AI.M Meeting today (365 chars)
Currently at message 1
```

The first time you use MM, or if you have somehow deleted your mail receiving file, you will see a message like:

```
?Can't open message file "<[USERNAME]>MAIL.TXT.1" - file does not exist
```

This means that you have not yet received any messages. A file will be created automatically when an incoming message is received.

To read your messages, type READ <CR> at the MM> prompt. This will cause the first message in your MAIL.TXT file to be displayed on the screen. Alternatively, if you have a numbered list of messages as shown above, you can type READ [NUMBER] <CR> to display only the specified message. After the first message is displayed, your prompt will be R>. This is the reply prompt. Here, you have some choices. If you wish to answer the message,



type `REPLY <CR>`. The `REPLY` function associates your reply with the original message; you do not need to type a new heading. Type in your message, then a `↑Z`. Type `SEND <CR>` at the `S>` prompt. Your message will be queued, and the `R>` prompt will reappear.

If you wish to forward the message to another user, type `FORWARD [USERNAME] <CR>` at the `R>` prompt. When you see the message instructions, add any text you wish to be sent with the forwarded mail and type `↑Z` to end the message. Type `SEND <CR>` at the `S>` prompt.

If you wish to remove the message you have just received from your `MAIL.TXT` file, type `DELETE <CR>` at the `R>` prompt. If you wish to leave the message in your file and go on to read your other messages, type `NEXT <CR>`, or just press `<CR>`. After you have seen the last message, the `MM>` prompt will reappear; to return to the `MM>` prompt without looking at all the messages you have received, type `QUIT <CR>` at the `R>` prompt.

When you are back at the `MM>` prompt, type `EXIT <CR>` to leave the `MM` program and return to the top level of `OZ`.

### 3.3 Using BABYL

#### a. The Basics

`BABYL`, like `MM`, is a program for sending and receiving mail. It is fairly easy to learn at a beginner's level, but it has sufficiently complex features to keep you interested as you get more experienced.

When you first use `BABYL`, you will need to initialize a `BABYL` mail receiving file. To do this, type `BABYL <CR>` at the `@` prompt. The following will appear:

```
There is no Babyl file  OZ:<[USERNAME]>[USERNAME].BABYL.O
Do you want to create one?   (Y or N)
```

Type `Y`. A series of questions will appear on the screen, together with suggested safe and appropriate answers in brackets. Type in the answers as suggested (they can be changed later when you know more about using `BABYL`). The incoming mail file should be the default file `<[USERNAME]>MAIL.TXT` – press `<CR>`. You will be asked whether the file will have an owner. Type `Y`. After you do so, you will see a line at the bottom of the screen asking for the owner's name. Enter your username and a `<CR>`.

After you have finished answering these questions, you will be in your new `BABYL` file. There will be a sample message displayed so that you can see the format. You will usually see the text of some item of mail on the screen when you enter `BABYL`. This will either be a letter you have not yet read, in which case the top line at the bottom of the screen will contain the word `unseen`, or an item of old mail which you have not deleted. If the text of

the message fills more than one screen, the word `--More--` will appear at the bottom. Hit `<space bar>` to see the rest.

Near the bottom of the screen will be information about what you are currently doing in BABYL and the status of your messages. It will look like the following:

```
Babyl (Message 1/2, unseen) OZ:<USERNAME>USERNAME.BABYL.1
```

```
Reading Babyl file OZ:<[USERNAME]>[USERNAME].BABYL.1
```

```
Reading Mail file OZ:<[USERNAME]>MAIL.TXT.0
```

```
Appending to Babyl file OZ:<[USERNAME]>[USERNME].BABYL.1
```

The commands for manipulating text in BABYL are very similar to those for EMACS, which you will learn about in Section 4. You may wish to return to the BABYL section after you have learned a bit about EMACS, and use MM in the meantime. The most basic EMACS editing commands used in BABYL are the following:

`<delete>` or `<rubout>` key – Delete the character preceding the cursor

`↑D` – Delete the character under the cursor

`↑F` – Move the cursor Forward one space

`↑B` – Move the cursor Backward one space

`↑N` – Move the cursor to the Next line

`↑P` – Move the cursor to the Previous line

## b. Sending Mail with BABYL

Within the BABYL program you can send mail by typing M. Most commands typed in BABYL do not need to be followed by a `<CR>`. After a pause, the following will appear at the top of your screen:

To:

`--Text follows this line--`

The cursor will be next to the colon. Type the username of the person to whom you wish to send the message. Then type `<CR>`, which will move the cursor down, creating a blank line. As in MM, you want to put the basic address information at the top of the letter. In BABYL you must type the headings yourself. Use the format below as an example:

```
To: username <CR>
```

```
cc: username <CR>
```

```
Subject: A Strange Event <CR>
```

If you wish to send a practice letter to yourself, put your username on both the To: and cc: lines. You don't need to put From: or a date; BABYL will do that automatically.

Now you are ready to write the text of your letter. Using the EMACS-like command ↑N, move the cursor below the line which says --Text follows this line--, and type in your message, separating the lines with a <CR>:

```
Today when I woke up, I discovered that I was in a room different
than the one in which I had gone to sleep. This confused me, and
I lay there awhile in bed trying to understand what had happened.
```

```
-Joe Username
```

After you finish entering your message, type ↑Z↑Z. There will be a delay, then the screen will read Queuing...Done, and the message (mail queued) will appear at the bottom. This means that your letter has been sent. If you decide you do not want to send the message, type M-X Abort <CR>.

### c. Using BABYL to Read Your Mail

When you enter the BABYL program, you can read any messages currently in your [USERNAME].BABYL file. The last message you have received will appear on the screen automatically when you enter BABYL. The message will be in the following form:

```
Date: Monday, 20 June 1983 14:23-EDT
From: [USERNAME]
To: [username]
cc: [username]
```

```
Today when I woke up, I discovered that I was in a room different
than the one in which I had gone to sleep. This confused me, and
I lay there awhile in bed trying to understand what had happened.
```

```
-Joe Username
```

```
Babyl (Message 2/2, unseen) OZ:<USERNAME>USERNAME.BABYL.1
```

```
Reading Babyl file OZ:<[USERNAME]>[USERNAME].BABYL.1
```

```
Reading Mail file OZ:<[USERNAME]>MAIL.TXT.0
```

```
Appending to Babyl file OZ:<[USERNAME]>[USERNAME].BABYL.1
```

You will also want to read the messages received prior to the one currently on your screen. Use the following commands:

- B - List messages in [USERNAME].BABYL file
- D - Delete the current message
- E - Expunge deleted messages from [USERNAME].BABYL file
- F - Enter editing mode
- G - Get any mail that has come in since you started the BABYL session
- N - Go to the Next message
- O - Put the message into another file; you will be prompted for file name.
- P - Go to the Previous message
- Q - Quit the BABYL program
- U - Undelete the previous message

#### d. Getting On-Line Help and Exiting BABYL

Like many programs, BABYL has a built-in help function. For a list of all BABYL commands, type ?. At the bottom of the screen you will see the message: Type a Babyl command character to describe, "\*" for all of them:. If you type \* after the colon, a list of commands will appear, together with a short explanation of what each one does. Many of these may seem obscure, but you will only need a few to get along at first.

To get out of the BABYL program, type Q. After a short pause, the © prompt will reappear at the left margin. If you encounter a problem in BABYL, and Q does not work, type ↑C. This will return you to the exec level of OZ and you can start over.

#### e. Using BABYL from within EMACS

You may also use the BABYL program to read your mail from within the EMACS editor. While in EMACS, type ↑X. At the bottom of your screen, you will see: C-X. At this point, enter R <CR>. This will put you into the BABYL program, and you can use the commands listed in the previous sections. To return to your EMACS file, type Q.

### 3.4 Maintaining Your Mail Files

Both MM and BABYL require that you maintain mail receiving files. MM causes a file to be created when a message is received, while BABYL asks you for information so that it can create a custom file for you. It is a good policy to delete all unimportant mail after you have finished reading it. If you neglect to do so at the time the message is read, you can edit your actual files, MAIL.TXT.1 (for MM) or [USERNAME].BABYL, to remove unwanted messages. (See Section 2 on files.) When your mail file starts to fill up, you may wish to copy it into a storage file (named something like OLDMAIL.APRIL), so that your mail receiving file is kept clean, and so that you can find old messages if you need to look at them.

Each time you send mail, a file will be created called MAIL.CPY which will contain a copy of the message you sent. Feel free to delete these files if you do not want to keep them.

### 3.5 Mail Short Cuts

There are three short cut ways to send mail without entering the MM or BABYL programs. They can be used right from the top level of OZ.

From the @ prompt, you can type MAIL [USERNAME] <CR>. You will then be asked for Subject: (there will be no To: or cc: prompts). Enter the subject, a <CR>, and type your message when you see the Msg: prompt. After you have completed the message, type ↑Z. You will receive a note telling you that your message has been queued, indicating that it has been put in the recipient's mail receiving file and will be seen when they next read their mail. Another short cut method is to type SEND [USERNAME] <CR> at the @ prompt. You will be prompted only for Msg:. Messages sent using this command will appear immediately on the recipient's video screen, and will not be put into their mail receiving file.

Because SEND messages appears immediately on the screen, you may occasionally miss a message, especially if you are in the middle of a process when the message arrives. To see your previous SEND messages, type HUH <CR> or WHAT <CR>. Typing WHAT [NUMBER] <CR> allows you to see multiple past SEND messages. Replace [NUMBER] with the number of messages you wish to view. SEND messages will be put into a file named SENDS.TXT. You may read, edit and delete this file as you choose.

The final short cut for sending mail, to be used for brief, urgent messages that are of general interest, is the SHOUT command. Type SHOUT <CR> at the @ prompt. You will be prompted for your message. Enter it (no longer than one line) and ↑Z. Messages sent using SHOUT appear immediately on everyone's screen. SHOUT messages should be used sparingly.

There is also a short cut method for reading messages. Type RMAIL <CR> at the top level. All awaiting messages will appear on your screen in succession. Hit <space bar> to see the remainder of long messages when you see --Pause-- or --More--. After all the messages have been scrolled on your screen, you will see at the bottom: Delete this mail? (Y or N). If you enter Y, all of the messages will be deleted. If you wish to save any of the messages, type N. This will cause the messages to be stored in your mail file for later reading and removal.

### 3.6 Using Mailing Lists

There are many mailing lists available to use when you are sending mail to a large number of people. Examples are "all-ai", "7-ai", "robotics-group", and "secretaries". You use these names just as you would a username. Make sure that you send to the major groups, such as "all-ai" only those messages that everyone must see.

There is a special mailing list called "bboard" for such things as seminar announcements, apartments for rent, etc. Everyone is on this list. You get messages sent to "bboard" when you log in, displayed under `--Msgs--`.

There is a file on OZ containing all the available mailing lists. It is called `MAILING-LISTS.TXT`, in the directory `<MAIL>`. There are mailing lists for a wide range of interests. Do not make changes to this list until you are experienced at manipulating files and using `EMACS` (see Sections 2 and 4). In the meantime, ask someone to put you on the mailing lists that you will need to be on immediately (i.e., those related to your work). Once you have mastered files and `EMACS`, feel free to put yourself on the lists you find interesting. There are instructions at the beginning of the `<MAIL>MAILING-LISTS.TXT` file that will tell you how to do this.

### 3.7 Long Distance: Using Networks

You may have noticed that the full mail address of OZ users, as it appears in message headings is: `[USERNAME]@OZ.AI.MIT.EDU`. This indicates that the user is logged into the computer named OZ, at the A.I. Laboratory, at MIT. To send messages to other OZ users, the username alone is sufficient, since OZ is assumed if no other computer is specified. To send messages to users on another computer at the lab, you must type `MAIL [USERNAME]@[COMPUTER NAME] <CR>`.

The suffix `.EDU` in the full address signifies that the computer is part of a nationwide network of universities and research institutions. To send mail to users on this network, use an address of the form `[USERNAME]@[COMPUTER NAME].[DEPARTMENT].[UNIVERSITY].EDU`. For example:

```
MAIL FRED@VAX.COQ-SCI.USC.EDU <CR>
```

indicates a user named "Fred", using the computer "VAX" in the Cognitive Science Department at the University of Southern California.

There are other national and worldwide networks which can be accessed from OZ. The most commonly used are `.ARPA` (the Department of Defense research network), `.BITNET`, and `.INTERNET`. Addresses for these networks are similar, but not identical, to those for the `.EDU` net.

### 3.8 A Practice Session

This practice session will show you how to use `MM`. This session assumes you are already logged on, and at the `exec` level of OZ. You will be sending messages to yourself, then reading them.

@

==> MM <CR>

?Can't open message file "<[USERNAME]>MAIL.TXT.1" - file does not exist

NOTE: This indicates that you currently have no messages in your MAIL.TXT file. To remedy this, send a message to yourself:

MM>

==> SEND <CR>

To:

==> [YOUR USERNAME] <CR>

cc:

==> <CR>

Subject:

==> Sample letter <CR>

Message (End with ESCAPE or CTRL/Z.

Use CTRL/B to insert a file, CTRL/E to enter editor, CRL/K to redisplay message, CTRL/L to clear screen and redisplay, CTRL/N to abort.):

==> This is a sample letter which you can mail to yourself to see how the MM program operates. Feel free to type anything you like here.

-Joe Username

↑Z

S>

==> SEND <CR>

username@OZ.AI.MIT #Chaos -- queued

MM>

NOTE: The following notice will appear after a pause:

[You have a message from USERNAME]

==> <CR>

MM>

==> READ <CR>

There is 1 additional message

N 1) 1-Jul To: username@OZ.AI.M Sample letter (335 chars)

Currently at message 1

Message 1 (335 characters):

Return-path: <[USERNAME]@MIT-OZ>

Mail-From: [USERNAME] created at 1-Jul-87 14:21:33

Date: 1 July 1987 1421-EDT

From: Joe J. Username <[USERNAME]@MIT-OZ>

Subject: Sample Message

To: <[USERNAME]@MIT-OZ>

This is a sample letter which you can mail to yourself to see how the MM program operates. Feel free to type anything you like here.

-Joe Username

R>

==> REPLY <CR>

Message (End with ESCAPE or CTRL/Z.

Use CTRL/B to insert a file, CTRL/E to enter editor, CRL/K to redisplay message, CTRL/L to clear screen and redisplay, CTRL/N to abort.):

==> Thank you for your sample letter!

↑Z

S>

==> SEND <CR>

username@OZ.AI.MIT #Chaos -- queued

NOTE: The message below will appear. Remember that you are sending mail back and forth yourself, so the messages appear immediately.

[You have a message from USERNAME]



==> <CR>

There is 1 additional message

N 2) 1-Jul To: username@OZ.AI.M Re: Sample letter (225 chars)  
Currently at message 2

MM>

==> READ <CR>

Message 2 (225 characters):

Return-path:<[USERNAME]@MIT-OZ>

Mail-From: USERNAME created at 1-Jul-87 14:23:30

Date:1 July 1987 1423-EDT

From: Joe J. Username <[USERNAME]@MIT-OZ>

To: <[USERNAME]@MIT-OZ>

Subject: Re: Sample Message

In-reply-to: Msg of 1 Jul 87 14:21 EDT from USERNAME

Thank you for the sample letter!

R>

==> DELETE <CR>

R>

==> NEXT <CR>

NOTE: Use this command repeatedly until the MM> prompt reappears.

MM>

==> EXIT <CR>

Expunging deleted messages.

@

## 4. EDITING WITH EMACS

### 4.1 Introduction

EMACS is an *editor*, a program you can use to enter, move, and manipulate text. It has many features that allow you to do advanced word processing. In this section, you will learn about the various commands needed to move and rearrange text, how to use M-X commands to perform special functions, and how to design keyboard macros to simplify your editing work.

The EMACS program has many commands, some of which perform quite complex functions. You will only need a small subset of commands to get started, and these are included in this guide. As you gain proficiency with EMACS, you should consult one of the available manuals to learn more. The **EMACS Manual for TWENEX Users**, by Richard M. Stallman, A.I. Laboratory Memo 555, © MIT (1983), is highly recommended, especially for its comprehensive command summary.

### 4.2 On-Line Help and the TEACH-EMACS Tutorial

EMACS contains extensive on-line documentation. The help character is ↑\_ (control underscore). When you type this command, you will see the help prompt: Doc (? for help): in the space at the bottom of your screen. Typing ?<CR> after the colon will give you some basic information. If you type a C at the colon, you will be prompted to enter a character command to be explained. If you type a D after the colon, you will be prompted for a function to be described. Type ↑X I to enter a file that contains a tree-structured listing of EMACS information (type ↑x to exit). There are more complex methods of getting help within EMACS; consult the **EMACS Manual** for more information.

There is also a training program called TEACH-EMACS which will take you through the basics of using the EMACS editor. This program contains a sample file for you to practice with. In order to use this program, type TEACH-EMACS <CR> at the © prompt.

### 4.3 Using Buffers

EMACS is designed to work with *buffers*. Buffers are unsaved files that contain work in progress. Normally when you are editing, you will be working with only one buffer (the *main buffer*), but you may create and work with as many buffers as you choose. The contents of a buffer will be lost when you logout (or if the system crashes) unless you explicitly save the buffer as a file using ↑S↑X. Buffers should not be confused with *minibuffers*, which are auxiliary buffers used for entering commands within EMACS. Use the following commands to list and manipulate buffers. Consult the **EMACS Manual** for a more detailed explanation of buffers and their uses.

`↑C↑B` - List all currently existing buffers (a \* means that the buffer has not been saved)  
`↑C B` - Set up a prompt to choose which buffer to view or use  
`↑C K` - Kill the specified buffer (default is the current buffer)  
`↑G` - Abort the last command set in motion  
`RESET EMACS <CR>` - Kill all existing buffers (execute at top level)

#### 4.4 Entering EMACS to Edit or Create a File

To enter a file, you need to call up the EMACS program. To do this, type `EMACS <CR>`. This initiates an EMACS job (assuming you do not have any other EMACS jobs currently active). When you do this, the following will appear at the top of your screen:

```
EMACS Editor, version 165 - type ^ (the help character) for help.
```

At this point type `↑X↑V`; you will see the following at the bottom of your screen:

```
Visit File (Default OZ:<[USERNME]>GAZONK.DEL.0):
```

Type the name of the file you wish to work with after the colon. If you want to create a new file, type the name you wish to give the file (see Section 2 on creating files). You may also type `↑X↑F` to find a file. Type the file you wish to locate after the colon.

Another way to enter the editor is to type `EMACS [FILENAME] <CR>` or `EDIT [FILENAME] <CR>` at the `@` prompt. If you type just `EMACS <CR>` or `EDIT <CR>` without specifying a file name, you will be returned to the last file you worked on; the most recently used file is kept in a buffer until you either `RESET EMACS` or evoke `EMACS` with a different file name.

#### 4.5 Using Windows Within EMACS

If you wish to work with two different files simultaneously, it is possible to create *windows*. Windows allow two files to appear on your screen, one above the other, and enable you to move between them to edit both files. To create a second window, type the command `↑C 2`. This will open a new window at the bottom half of the screen. The cursor will appear in the new window. You can edit a new file in this window by using the `↑X↑F` command (described in the previous section). Move back and forth between the two windows by typing `↑C 0` (for "Other"). To resume editing on the full screen, type `↑C 1`. This will cause whichever window you were in when you issued the command to fill the entire screen, and the other window will disappear. Be sure to save whatever work you did in that window before deleting it (see Section 4.12 on saving files).

## 4.6 Basic Editing Commmands

### a. The Status Line

Whenever you are editing an EMACS file, you will see a *status line* at the bottom of your screen that will give you information about your current editing job. This line will be similar to the following:

```
EMACS (Fundamental) Main: OZ:<[USERNAME]>NAME-OF-FILE.TXT (15) --50%-- *
```

The first part tells you that you are currently in the EMACS program. The word within parentheses indicates the mode you are currently in. This will usually be (Fundamental) or (Fundamental Fill). The third part of the status line shows the full name of the file you are working on. The number following in parantheses is the current version number. The next field will be a percentage, and will let you know your current location within the file. For example, --50%-- means your cursor is located half way through the file. The \* indicates that changes have been made since you last saved the file.

### b. Moving Within the File

Once you have entered the editor, you will need at know how to move around in the file. The most frequently used cursor movement commands are the listed below:

- ↑V - Move down one screen of text
- M-v - Move up one screen of text
- M-< - Move to the beginning of file
- M-> - Move to the end of file
- ↑P - Move to the Previous line of text
- ↑N - Move to the Next line of text
- ↑F - Move Forward one letter
- ↑B - Move Backward one letter (same as <back space> key)
- M-F - Move Forward one word
- M-B - Move Backward one word
- ↑A - Move to beginning of the current line
- ↑E - Move to End of the current line
- M-[ - Move to beginning of the current paragraph
- M-] - Move to end of the current paragraph
- ↑U [LINE NUMBER] ↑N - Move to the specified line number.

More complex movement commands are explained in the **EMACS Manual**.

### c. Inserting and Transposing Text

To enter text into a new file or to add to an old one, use the movement commands to move the cursor to where you wish to put the new text and begin typing. Use the `↑O` command to open up a new line at the cursor. There is no special command needed to enter insert mode.

If you make a mistake, you may use the following commands to reverse the order of letters and words:

`↑T` - Transpose two letters  
`M-T` - Transpose two words

### d. Deleting and Undeleting Text

You can use the `<delete>` or `<rubout>` key to delete the character preceding the cursor. The commands listed below are for more complex deletions:

`↑D` - Delete character under the cursor  
`↑K` - Delete ("Kill") the current line from the cursor to the end  
`M-D` - Delete one word forward  
`M-<del>` - Delete one word back  
`↑X <del>` - Delete one sentence back

If you make a mistaken deletion, use the command `↑Y` to "Yank", or reinstate, the text.

## 4.7 Searching for Strings of Text

EMACS has commands to locate specific strings within the text file. A *string* is any piece of text; it can include letters, digits and punctuation, and is of arbitrary length.

`↑S` - Search forward in the file  
`↑R` - Search backward in the file

When you enter one of these commands, you will see a line at the bottom of the screen that will say `I-Search:` or `Reverse I-Search:` Enter the desired string after the colon and press `<CR>`. Pressing `↑S` or `↑R` repeatedly will find subsequent or previous occurrences of the same string. If the string is not found, you will see the message: `Failing I-Search`, and your cursor will be moved to the string that is most similar to the one you requested. To end the search, type the `<escape>` key.

## 4.8 Moving and Copying Blocks of Text

You can move and copy blocks, or *regions*, of text by marking the text to be moved, deleting it, and reinstating it at the desired location. The procedure is as follows:

**M-C** or **↑<space bar>** – Mark the beginning of the desired region

Use the cursor movement commands to move to the end of the desired region

**↑W** – Delete (“Wipe”) the text within the marked region; this is saved in an EMACS memory buffer.

**M-W** – Alternatively, add the marked text to the buffer to be moved, but keep a copy in the original location as well.

Use the movement commands to put the cursor where you want the text to appear

**↑Y** – Reinstale (“Yank”) the text at the new location.

You can use the **↑Y** command repeatedly to place the wiped text in more than one place. To copy text, follow the instructions for marking the text as shown above, then use **↑W** to wipe the text. Enter **↑Y** immediately to reinstale the text at its original location, then move to the new location where you wish to place a copy of that text and type **↑Y** again. You will then have copy of the desired text in both the original and the new location.

There are also commands you can use to mark paragraphs, pages and buffers. Consult the **EMACS Manual** to learn more about marking and moving regions of text.

## 4.9 Some Special Character Commands

Below are some commands you can use to perform special operations within EMACS. They are known as *character commands* because the operations are performed when you type in special characters. The following are only a few of the available commands. See the **EMACS Manual** for more.

**M-L** – Convert the current word to Lower case letters

**M-Q** – Reformat the current paragraph

**M-S** – Center the text preceding the cursor

**M-U** – Convert the current word to Upper case letters

Use the above commands to change the appearance of your text file.

**M-\_-** – Underline the word preceding the cursor

Move to the end of the word you want underlined and enter the command. Control characters will be inserted before and after the underlined word to let the computer know that it should be underlined when printed. This will look like: **␣word␣**. The actual underscore characters will not appear on the screen.

**M-\$** – Check the spelling of the previous word

Move the cursor so that it is on or after the word you wish to check and enter the command above. If the word is correct, you will see: **Found it.** at the bottom of your screen. If the word is incorrect, suggested correct spellings will appear at the top of the screen. You can change the word by typing in the digit listed in front of the correct spelling. If you wish to leave the word as it is, type **<space bar>**. If a word is unknown to the computer (for example a proper name), the message: **Word not found.** will appear at the bottom of the screen.

#### 4.10 Extended (M-X) Commands

There is a special feature of EMACS that allows you to enter specialized commands and perform complex functions that cannot be done using the character commands alone. These are called *extended* (or *M-X*) *commands*. Below are some basic M-X commands. To execute these commands, type **M-X** anywhere within the file. Your cursor will reappear at the bottom of the screen; here you can enter the command word(s). To abort an M-X command if you change your mind, type **↑G** (twice if necessary). The command **M-X Apropos** will give you more information about extended commands.

**M-X AUTO FILL <CR>** – Switch on and off Auto Fill Mode

EMACS can be set so that the text wraps around to the next line as you type. Typing this command alternately turns **Auto Fill Mode** on and off. If it is on, the status line at the bottom of your screen will say **(Fundamental Fill)**.

**M-X CHECK MAIL <CR>** – List the mail currently waiting to be read

**M-X CORRECT SPELLING <CR>** – Check and correct the spelling of the entire file

This command invokes the **ISPELL** program, which will search through the file, find misspelled words, and ask you whether you wish to change them. Enter this command at the beginning of the file to be checked. The **ISPELL** program works in a manner similar to the **M-\$** command for individual words. Suggested correct spellings will be listed on the screen. Fix the misspelled word by typing the digit that corresponds to the spelling you prefer. If the program makes no suggestions, or if you do not like the ones it makes, type **R** to replace the misspelled word with one of your own choosing. Press **<space bar>** to leave the word as it is

**M-X DIREDD <CR>** – List the files in your directory from within EMACS.

**M-X LISP MODE <CR>** – Set <tab> key to properly indent LISP programs

Typing this command alternately turns LISP Mode on and off. When on, using the <tab> key produces indentation similar to pretty-printing and matching parentheses will be indicated. If LISP Mode is on, the status line of file will say (LISP).

**M-X REAP <CR>** – Delete all but the two most recent versions of each file

This command can be used to clean up your directory from within EMACS. You must still return to the exec level to expunge the files, as shown in Section 2.

**M-X REPLACE STRING <CR>** – Replace all occurrences of a string within the file

**M-X QUERY REPLACE <CR>** – Replace specific occurrences of a string within a file

These commands will locate occurrences of a string and replace them with a substitute string. After typing either of the commands above, press the <escape> key. A \$ will appear after the command name. Type the string you wish to replace after the \$ and hit <escape> again. Another \$ will appear. Now type the string to be substituted. Hit <escape> once more, and then <CR>. If you use REPLACE STRING, all instances of the string throughout the file will be replaced immediately. If you use QUERY REPLACE, you will be presented in turn with each occurrence of the original string. Typing <space bar> causes that instance of the string to be replaced with the substitute string. Typing <delete> will leave it as it is. ↑G will stop the query replace process.

**M-X LOAD LIBRARY SORT <CR>** – Load the sort routine library

**M-X SORT LINES <CR>** – Alpha-numerically sort the file by lines

**M-X SORT PARAGRAPHS <CR>** – Sort the file by paragraphs

The LOAD command is used to access a library of sorting routines maintained by EMACS. To use the sort commands, set a mark by placing the cursor where you wish to begin the sort. Move the cursor using ↑N to the point at which you wish to end the sort. Then type the appropriate command to sort either by lines or by paragraphs. The sort will be alpha-numeric, numbers preceding letters. Use ↑X↑S to save the sorted version.

**M-X UNDO <CR>** – Reinstate the most recently killed material

If you mistakenly delete some text, this command causes the most recently killed text to be put back at the location from which it was removed. Enter the command and press <CR>. The prompt at the bottom will read: Undo the last kill (Y or N)?. Type Y to reinstate the text.

Consult the EMACS Manual for a full list of M-X commands, along with a more detailed description of how to use them.



#### 4.11 Using Keyboard Macros

A *keyboard macro* is a special function you can define to abbreviate a sequence of commands. This will allow you to perform multi-command operations with a single character command. The procedure on the following page will show you how to create simple macros. Such macros can also be named or stored as permanent user-defined commands. See the **EMACS Manual** for more details.

↑X ( – Begin defining a keyboard macro

Type the commands you want to become a part of the current macro

↑X ) – End the definition of the current keyboard macro

↑X E – Execute the most recently defined macro (this may be used repeatedly)

↑U [NUMBER] ↑X E – Execute the macro the specified number of times

↑G – Abort the current macro

#### 4.12 Saving the File and Exiting EMACS

When you are finished editing a file, save it by typing ↑X↑S. A new version of the file will be created that contains the changes you have just made, and the following will appear at the bottom of your screen:

```
Written: OZ:<[USERNAME]>NAME-OF-FILE.TXT.15
```

You may also save the changes you have made to any other file you specify. Type ↑X↑W from within EMACS, and enter the desired file name after the colon when prompted to do so.

It is recommended that you save your files frequently, every page or so. If the system crashes, any editing work done on the file before the last save is safe; you may lose changes made after that time. The \* in the status line indicates that there are modifications that have been made to the file but not yet saved. If no changes have been made since the last save, you will see: (No changes need to be written) at the bottom of your screen when you attempt to write the file.

To exit the editor, type ↑X↑Z (↑Z↑Z will also work). To exit the editor without saving changes, or to return to the top level of OZ at any time, type ↑C. When your editing session is finished, type RESET EMACS <CR> to kill any currently active EMACS jobs.

#### 4.13 A Practice Session

There is a file in the directory <LIZ.GUIDE> that contains a file for you to practice on. If you went through the Practice Session at the end of Section 2, you will have already copied this file into your own directory. If not, type:

COPY <LIZ.GUIDE>EXAMPLE.TXT <[YOUR DIRECTORY]>EXAMPLE.TXT <CR>.

The actual text, as it appears in the file EXAMPLE.TXT, appears on the following page:

```
\typesize=11pt
\hsize=6.0in
\vsize=9.0in
\centerline{ \bigsize \bf ‘‘GENIUS’’}
\ medskip
```

We naturally admire our Einsteins, Shakespeares and Beethovens -- and we wonder if machines could ever create such wondrous theories, plays and symphonies. Most people think that accomplishments like these require ‘‘talents’’ or ‘‘gifts’’ that cannot be explained. If so, then it follows that computers can’t create such things -- since anything machines can do can be explained. But why assume that what our greatest artists do is very different from what ordinary people do -- when we know so little about what ordinary people do! Surely it is premature to ask how great composers write great symphonies before we know how ordinary people think of ordinary tunes. I don’t believe there is much difference between normal and ‘‘creative’’ thought. Right now, if asked which seems the more mysterious, I’d have to say the ordinary kind.

We shouldn’t let our envy of distinguished masters of the arts distract us from the wonder of how each of us gets new ideas. Perhaps we hold on to our superstitions about creativity in order to make our own deficiencies seem more excusable. For when we tell ourselves that masterful abilities are simply unexplainable, we’re also comforting ourselves by saying that those superheroes come {\ it endowed} with all the qualities we don’t possess. Our failures are therefore no fault of our own, nor are those heroes’ virtues to their credit either. {\ If it isn’t learned, it isn’t earned}.

```
\medskip
\noindent
({ \ it Excerpted with permission from:} Marvin Minsky,
{ \ bf The Society of Mind}, Simon and Schuster, New York, NY, 1986, p. 80.))
```

**NOTE:** The *backslash commands* at the beginning of the file are for the text formatting program YTeX. In Sections 5 and 6, you will learn what these commands mean, and will be shown how to run the text formatting program and print out the results.

Practice using EMACS by editing the file EXAMPLE.TXT:

⓪

==> EMACS

EMACS Editor, version 165 - type ^\_ (the help character) for help.

==> ↑X↑V

Visit File (Default OZ:<[USERNME]>GAZONK.DEL.0):

==> EXAMPLE.TXT <CR>

**NOTE:** This will put you into the EXAMPLE.TXT file. As you will notice, the file contains some spelling mistakes. Practice moving around in the file and correcting these mistakes using the commands you have learned in this section. Since you have made your own copy of EXAMPLE.TXT, do not worry about making mistakes. If you somehow ruin or delete your EXAMPLE.TXT file, or if you wish to start over with a fresh version, use the COPY command shown above to make a new copy in your directory. To save your file and end the editing session, type from within the file:

==> ↑X↑S

Written: OZ<[USERNAME]>EXAMPLE.TXT.2

==> ↑X↑Z

⓪

==> RESET EMACS <CR>

⓪

## 5. TEXT FORMATTING WITH YTeX

### 5.1 Introduction

YTeX is a program which will read a file that you have prepared using EMACS and translate it into a set of instructions which will cause the printer to output the file in the style you have specified. This process is known as *text formatting*, or *typesetting*. Although it is possible to print a “raw” EMACS text file directly, formatted files have a much more pleasing appearance and allow a wider range of style options.

YTeX is not the only text formatting program available to you on OZ. Another popular formatting program is LaTeX. This guide will discuss only the YTeX program, because it is somewhat easier and more widely used. LaTeX is quite similar to YTeX, but includes some options that YTeX does not offer, such as table of contents and bibliographic database preparation, and a package called SliTeX for producing text for transparencies.

This guide will show you some of the basic features of YTeX, and the commands you will need to format simple documents. For an overview of YTeX, a good resource is *How to Use YTeX*, by Daniel Brotsky, A.I. Laboratory Working Paper 273, © Massachusetts Institute of Technology (1986). For complex typesetting of equations, see *The TeXbook*, by Donald Knuth, © American Mathematical Society (1986). Although this book is based on the original TeX program, you should find it quite helpful. If you prefer to use LaTeX, refer to the manual *LaTeX Document Preparation System: User's Guide and Reference Manual*, by Leslie Lamport, © Addison-Wesley (1986). There are also a number of A.I. Laboratory memos and working papers that deal with these programs (see Appendix C). The text formatting programs are frequently modified and updated, so these memos are an important source of up-to-the-minute information.

YTeX will probably seem quite confusing to you at first, especially if you are accustomed to working with a dedicated word processing system. Remember that you only need to be familiar with a small subset of commands in order to get started. A good way to learn about YTeX is to compare a .YTEX file to the final formatted document and notice what does what. The file <LIZ.GUIDE>IDIOTS-GUIDE.YTEX contains the text of this guide in YTeX form; you can compare this to the printed copy of the guide to see how the various stylistic features were produced.

### 5.2 Using YTeX: The Basic Idea

YTeX commands, also called *backslash commands*, are commands inserted throughout the text file to tell the printer what the final output should look like. These commands always begin with the character \, and should be typed in lower-case letters. Files that contain commands for YTeX will usually end in the suffix .YTEX. Once you have prepared

your .YTEX file, you can run the YTeX program on it to produce a new file that contains instructions that can be interpreted by the printer.

In the course of preparing your .YTEX file, you may wish to insert *comments* into the file as notes or reminders to yourself. This can be done anywhere within the file. To enter a comment, type the character % followed by the text of the comment.

### 5.3 YTeX Opening Commands

There are certain YTeX commands that must be included at the beginning of every .YTEX file. These commands tell the printer the general format of the document.

#### a. Setting Sizes

At the beginning of the file, you should enter the commands that will set the size of the type and of the margins that will appear throughout the formatted document:

```
\typesize=11pt
\hsize=6.0in
\vsiz=9.0in
```

The first command above indicates the size of the type you wish to use. 10pt is the size used by most typewriters; 11pt and 12pt are slightly larger.

The \hsize (horizontal size) command sets the width of the text to be printed. The \vsiz (vertical size) command sets the length of the text on the page. If you are using the usual 8.5 x 11 inch paper, an \hsize of six inches (6.0in) and a \vsiz of nine inches (9.0in) will produce attractive margins.

#### b. Inputting Auxiliary Files

If you frequently produce complex documents which have a similar format, you may wish to set up a file that contains the basic formatting commands you use instead of entering them anew each time you create a .YTEX file. This file, called <[USERNAME]>MACROS.YTEX or <[USERNAME]YTEX.LOCAL, can contain commands for sizing, page numbering, section numbering, etc. Look in the file <LIZ.GUIDE>MACROS.YTEX to get an idea of what is contained in such a file. If you wish, you can copy this into your directory to create your own MACROS.YTEX file.

When you run the YTeX program on a specific .YTEX document file, you will want the program to read the commands in the MACROS.YTEX file. To do this, use the \input command:



When you skip a line between two portions of text, the YTeX program will assume that you are starting a new paragraph. It will insert a blank line into the finished output and indent the following line five spaces. If you do not want your text to be indented, enter the command below on the line before the text:

```
\noindent                %Do not indent the next line
```

If you wish to center a line of text, use the following command:

```
\centerline{ text to center}    %Center a line of text
```

Put the text to be centered within the brackets.

There are two ways to create underlined text. The command:

```
\underbar{ text to underline}    %Underline text
```

will cause the text within the brackets to be underlined.

The command:

```
\line{ \hrulefill}              %Draw a line
```

will draw a line across the page from margin to margin. This can be used under a line of text or by itself.

YTeX has commands that automatically number chapter and section headings in outline form, and print the headings in bold face type and graduated sizes. Typing the commands:

```
\chapter{ This is Chapter One}    %Begin a chapter
\section{ This is the First Section} %Begin a section
```

will produce the formatted output:

## 1. This is Chapter One

### 1.1. This is the First Section

Other commands are available in YTeX that will produce inverted paragraphs, bulleted paragraphs and more. Refer to *The TeXbook* for details.

#### b. Spacing Commands

When you are editing, inserting blank lines into the .YTEX file will not cause such spacing to appear in the finished output. The ways to make blank lines appear in the formatted version are shown below:

<code>\smallskip</code>	<code>%Small Skip - about one half line skipped</code>
<code>\medskip</code>	<code>%Medium Skip - about one line skipped</code>
<code>\bigskip</code>	<code>%Large Skip - about two lines skipped</code>

Exact spacing will depend on the typesize and font you are using. You may have to experiment to achieve the spacing you desire.

To skip larger chunks of space, you can use the `\vskip` command. This command works with measurements in inches or millimeters:

<code>\vskip 0.5truein</code>	<code>%Skip one half inch</code>
<code>\vskip 4.0mm</code>	<code>%Skip four millimeters</code>

To double space your entire file or a part of it, use the following commands:

<code>\doublespace</code>	<code>%Begin double spacing</code>
Text to be double spaced	
<code>\singlespace</code>	<code>%End double spacing</code>

Normally, the YTeX program will break your document into pages properly. However, if you wish to override the YTeX pagination, use the command:

<code>\vfil\eject</code>	<code>%Start a new page</code>
--------------------------	--------------------------------

#### c. Font Changing Commands

One of the nice features of YTeX is that it allows you to use a variety of different fonts to improve the appearance of your documents.



For example, if you have set your typesize to 11pt, you may wish to have certain lines, such as titles, printed in a large or larger size type. To do this use the commands:

```
{\bigsize [text]}           %Large size text
{\biggsize [text]}         %Larger text
```

These commands print text in increasingly larger type fonts. Put the text to be enlarged after the command within the brackets (do not type the [ ]). You may have to experiment get the size you want.

If you wish to emphasize a string of text, you can put it in *italics*, **bold face**, *slanted type* or **typewriter font**. Use the following commands, replacing [text] with the text to be emphasized (omit the [ ]):

```
{\it [text]}               %Italics
{\bf [text]}               %Bold face
{\tt [text]}               %Typewriter font
{\sl [text]}               %Slanted type
```

#### d. Special Characters

As you will see when you begin to format equations, many of the specialized YTeX commands contain normally used symbols. For example, the equation delimiter is \$. If you wish to actually print out a \$, you will need to precede it with a backslash (\\$). This also true for the symbols #, & and others.

### 5.5 Formatting Equations

Using YTeX you will be able to typeset most complex equations. As the sample equations on the next page demonstrate, available features include Greek letters (upper and lower case); varying sizes of parentheses, brackets, and braces; subscripts and superscripts; and symbols for derivative, sum, square root, and vector, among others.

#### a. Horizontal Mode

There are two modes of equation formatting in YTeX. The first is called *horizontal mode*, also known as *text style*. In this mode, numbers, symbols, and short equations are inserted within a line of normal text. When typing, the equation should be preceded and followed by single dollar signs. Letters within the dollar signs will be italicized. For example, if you type:

If  $m=2$ , then the solution is  $J_2(f)$ .

the formatted output will be:

If  $m = 2$ , then the solution is  $J_2(f)$ .

Horizontal mode can also be used to number references, as the following example shows. If you type:

This fact was also pointed out by Winograd<sup>[15],[23]</sup>.

the processed output will be:

This fact was also pointed out by Winograd<sup>[15],[23]</sup>.

## b. Vertical Mode

The second available math mode is called *vertical mode*, or *display style*, and is used to produce more complex equations that are set off from the text, as shown by the sample equations on the next page:

$$\Omega(\mu, \nu) = \frac{s_3 - ps_1 - qs_2}{\sqrt{1 + p + q^2}} \quad (1)$$

$$J_m(f) = \int \int_{-\infty}^{+\infty} \sum_{\nu=0}^m \binom{m}{\nu} \left( \frac{\partial^m f}{\partial x^\nu \partial y^{m-\nu}} \right)^2 dx dy \quad (2)$$

$$\left( -\frac{4f}{4 - (f^2 + g^2)}, -\frac{4g}{4 - (f^2 + g^2)}, 1 \right) \cdot \vec{s} \quad (3)$$

Equations in vertical mode are delimited by double dollar signs in the .YTEX file. These equations may extend over several typed lines. As an example of how to use this mode, the input below was used to produce Equation (1) above:

```
$$ \Omega(\mu, \nu) = \{ \{ s_3 - ps_1 - qs_2 \} \over
\{\sqrt{1 + p + q^2}\} \} \} \eqno(1.1)$$
```

### c. Some Basic Equation Commands

There are too many equation typesetting commands to list in this manual, and most of them are only used for specialized purposes. Some of the more basic commands are shown below. All of these may be used in either horizontal or vertical mode. See *The TeXbook* for a complete listing of the math mode commands and how to use them.

INPUT	FORMATTED OUTPUT	
<code>\delta</code>	$\delta$	%Lower case Greek letter
<code>\Delta</code>	$\Delta$	%Upper case Greek letter
<code>\int</code>	$\int$	%Integral sign
<code>\infty</code>	$\infty$	%Infinity sign
<code>x^2</code>	$x^2$	%Superscript
<code>y_3</code>	$y_3$	%Subscript
<code>\sqrt{2}</code>	$\sqrt{2}$	%Square root
<code>\vec z</code>	$\vec{z}$	%Vector
<code>x\le y\ne z</code>	$x \leq y \neq z$	%Less than or equal to; not equal

### 5.6 Creating Tables

YTeX will allow you to construct fairly intricate tables. A normal table will fit entirely on one page. To create this type of table, enter the command `\begin table`. A table that extends over page breaks is called an *open table*. Begin this type of table by typing `\begin opentable`. Directly after either of these opening commands will be an indicator of the table format, appearing in brackets (`[]`). Indicators to use are `l`, `c`, or `r`. These indicate that the table columns should be left-justified, centered, or right-justified, respectively. The number of indicator letters in the brackets will control the number of columns in the table. For example, `[lll]` will create three left-justified columns. When typing the actual table text, use the `&` symbol to indicate that you wish to begin the next column. A `\cr` is used at the end of each completed line of text. If used by itself, it will cause a line to be skipped. Use the command `\end table` or `\end opentable` to resume normal text editing. The following commands will create a simple table. See *The TeXbook* for more detail.

```

\line{\hrulefill}
\begin table [lcr]                                %Begin a table
COURSE TITLE &INSTRUCTOR &NUMBER \cr

```

```

\cr
Introduction to Artificial Intelligence &B. Horn &6.123 \cr
Visual Information Processing &T. Poggio &9.380 \cr
Seminar in Robot Hand Control &J. Hollerbach &9.393 \cr
Laboratory in Cognitive Science &E. Hildreth &9.888 \cr
Lisp for Linguists &M. Johnson &24.911 \cr
Graduate Topics in Linguistics &N. Chomsky &24.999 \cr
\endtable                                %End table mode
\line{\hrulefill}

```

The resulting table will look like:

COURSE TITLE	INSTRUCTOR	NUMBER
Introduction to Artificial Intelligence	B. Horn	6.123
Visual Information Processing	T. Poggio	9.380
Seminar in Robot Hand Control	J. Hollerbach	9.393
Laboratory in Cognitive Science	E. Hildreth	9.888
Lisp for Linguists	M. Johnson	24.911
Graduate Topics in Linguistics	N. Chomsky	24.999

## 5.7 Closing Commands

When you have finished entering text and are ready to end your .YTEX file, enter the following commands:

```

\filpage                                %Fill the current page
\end                                    %End the .YTEX file

```

If you wish to print out only part of your .YTEX file, place the \end command where you wish to stop printing.

## 5.8 Running the YTeX Program and Debugging the Input File

### a. Running YTeX

In order to translate the commands you have entered in the .YTEX file into commands that can be understood by the printer, it is necessary to run the YTeX program on the file.

To do so, type `YTEX [NAME-OF-FILE].YTEX <CR>`. This will initiate a YTeX job. As YTeX runs, you will see the current status of the program on your screen. The program will let you know what file it is processing, what input files it is using, and where it currently is in the file. Chapter and section numbers will be listed by name as they are processed, and page numbers will appear in square brackets, as shown below:

```
This is TeX, MIT Version 2.2 (preloaded format = ytex-cm 87.6.24)
(OZ:<[USERNAME]>[NAME-OF-FILE].YTEX.22 { YTEX version 2.0} { Nofixes}
{ typesize = 11.0pt} (OZ:<[USERNAME]>MACROS.YTEX.6) { Main title} [1]
[2] { Chapter: \ n This is Chapter One} [3] { Section: \ n This is
the First Section} [4] [5] [6] [7]
```

```
Output written on OZ:<[USERNAME]>[NAME-OF-FILE].DVI.1 (7 pages, 4136 bytes).
Transcript written on OZ:<[USERNAME]>[NAME-OF-FILE].LST.1.
```

#### b. The .DVI and .LST Files

As you can see above, the YTeX program produces an output file called `[NAME-OF-FILE].DVI`. This is the file that contains the instructions for the printer. All print files output by YTeX end with the suffix `.DVI`. The `.DVI` file is made up of unreadable control characters that will be interpreted by the printer. It does not contain the output as it will appear when printed. You can see how the file will look when printed if you have a terminal that supports `.DVI` previewing.

The YTeX program also produces a file that contains a transcript of the running of the program, including error messages. This is the file called `[NAME-OF-FILE].LST`. All transcript files end in the suffix `.LST`.

Each time you rerun the YTeX program on the same `.YTEX` file, the `.DVI` and `.LST` files will be updated and their version numbers incremented. After you have printed the final document with no errors, the `.DVI` and `.LST` files may be deleted, since the text is saved in the `.YTEX` file and can be reformatted if needed.

#### c. Debugging the .YTEX File

The YTeX status report shown on the previous page is what will appear if your `.YTEX` file contains no errors. If there is an error in the input file, the YTeX program will stop when the error is encountered and tell you what is wrong, as shown below:

```
This is TeX, MIT Version 2.2 (preloaded format = ytex-cm 87.6.24)
(OZ:<[USERNAME]>[NAME-OF-FILE].YTEX.22 { YTEX version 2.0} { Nofixes}
{ typesize = 11.0pt} (OZ:<[USERNAME]>MACROS.YTEX.6) [0] [1]
```

```
! Undefined control sequence.
1.55 \\
      bigskip
?
```

Many of the Y<sub>T</sub>eX error messages are cryptic, but with practice you will learn to interpret most of them. In this case `!Undefined control sequence` means that Y<sub>T</sub>eX did not understand a command. The line beginning 1.55 tells you that the error occurred on line number 55 of the file. As you can see, this line is broken after the `\\`. This is the point at which the error occurred. In this case, perhaps due to a typing error, the illegal sequence `\\bigskip` was entered instead of the legal command `\bigskip`. At this point, the easiest thing to do is to re-edit the `.YTEX` file to fix the error. To exit the Y<sub>T</sub>eX program, type `X` after the question mark. The following will appear:

```
No pages of output.
Transcript written on OZ:<[USERNAME]>[NAME-OF-FILE].LST.1.
@
```

Now you can re-enter EMACS to edit the `.YTEX` file. To do this just type `EMACS <CR>` at the `@` prompt if the `.YTEX` file was the last one worked on and is still in the buffer. If not, type `↑X↑F`, then enter `[NAME-OF-FILE].YTEX <CR>` after the colon.

To get to the line that contained the error, move the cursor to the beginning of the file and type `↑U`. Your cursor will reappear at the bottom of the screen. Here, type the number of the line on which the error occurred (shown in the Y<sub>T</sub>eX error message), and press `↑N`. The cursor will then be moved to the line that contains the error. You can now fix the error, save the file, and run the Y<sub>T</sub>eX program again. You may have to do this a number of times before the file is error-free.

If you get an error message you do not understand, press `<CR>` at the question mark following the message. The Y<sub>T</sub>eX program will usually skip over the error and continue processing the input file. Most often Y<sub>T</sub>eX will produce a `.DVI` file even if the input file contains errors. If you print the `.DVI` file and look at the output, you may be able to determine what the problem was from the appearance of the printed document. (Printing will be covered in Section 6.)

## 5.9 A Practice Session

There are a number of ways to practice using the Y<sub>T</sub>eX program. First, you can look at the file `<LIZ.GUIDE>IDIOTS-GUIDE.YTEX`, compare it with this guide, and see what commands produced what features. (Please do not make changes to this file.)

Second, you can enter your file <[USERNAME]>EXAMPLE.TXT, which you copied and worked with in previous Practice Sessions, and practice using some of the new commands you have learned. You may wish to try changing the commands in the file to vary size, font, etc. Run the YTeX program and see if it accepts your changes. If the fonts are unchanged, and if you fixed all the spelling errors during the Section 4 Practice Session, the file (as copied from <LIZ.GUIDE>EXAMPLE.TXT) will produce the following output when formatted:

## “GENIUS”

We naturally admire our Einsteins, Shakespeares and Beethovens – and we wonder if machines could ever create such wondrous theories, plays and symphonies. Most people think that accomplishments like these require “talents” or “gifts” that cannot be explained. If so, then it follows that computers can’t create such things – since anything machines can do can be explained. But why assume that what our greatest artists do is very different from what ordinary people do – when we know so little about what ordinary people do! Surely it is premature to ask how great composers write great symphonies before we know how ordinary people think of ordinary tunes. I don’t believe there is much difference between normal and “creative” thought. Right now, if asked which seems the more mysterious, I’d have to say the ordinary kind.

We shouldn’t let our envy of distinguished masters of the arts distract us from the wonder of how each of us gets new ideas. Perhaps we hold on to our superstitions about creativity in order to make our own deficiencies seem more excusable. For when we tell ourselves that masterful abilities are simply unexplainable, we’re also comforting ourselves by saying that those superheroes come *endowed* with all the qualities we don’t possess. Our failures are therefore no fault of our own, nor are those heroes’ virtues to their credit either. *If it isn’t learned, it isn’t earned.*

*(Excerpted with permission from: Marvin Minsky, The Society of Mind, Simon and Schuster, New York, NY, 1986, p. 80.)*

Third, run through the Practice Session below to create a .YTEX file that will produce a short letter when formatted.

Ⓞ

==> EMACS LETTER.YTEX <CR>

[EMACS 0]

EMACS (Fundamental) Main: OZ:<[USERNAME]>LETTER.YTEX (0) \*  
(New file)

```

==> \typesize=11pt           %Set typesize
      \input <liz.guide>macros.ytex %Input local YTeX command file

      \def\AIL{ Artificial Intelligence Laboratory} %User - defined macro

      \nopagenumbers          %Do not number pages of letter
      \noheaders

      \
      \bigskip                %Start six spaces down on page
      \bigskip
      \bigskip
      \beginnofill           %Put address heading in block style

13 July 1987
      \medskip                %Skip some space

Professor T. MacDuff
Department of Computer Science
University of Rochester
Rochester, NY 14627

      \bigskip                %Skip more space

Dear Professor MacDuff:
      \medskip
      \endnofill              %End non-filled mode

      \noindent               %Do not indent paragraph

I would be happy to have you visit our laboratory the next time you
are in Boston. I will be out of town until {\it August 16}, but any
time after that would be fine. Let me know when you have finalized
your plans.

      \medskip                %Skip space
      \beginnofill           %Enter no fill mode again

Yours sincerely,
      \vskip .75in            %Skip 3/4 inch
Professor T. MacBeth
MIT \AIL                     %Use of abbreviation macro
      \bigskip
TM:lh
      \endnofill             %End no fill mode

      \filpage                %Fill page

```



```

\end                                %End .YTEX file
↑X↑S
↑X↑Z

```

Written OZ:<[USERNAME]>LETTER.YTEX.1

@

```
==>  YTEX LETTER.YTEX <CR>
```

This is TeX, MIT Version 2.2 (preloaded format=ytex-cm 87.6.24)

```
(OZ:<[USERNAME]>LETTER.YTEX.1 { YTEX version 2.0} { No fixes} { \typesize=11.0pt}
```

```
(OZ:<LIZ>MACROS.YTEX.6)
```

! Undefined control sequence.

```
1.13 \bginnofill
```

?

```
==>  X <CR>
```

No pages of output.

Transcript written on OZ:<[USERNAME]>LETTER.LST.1.

@

```
==>  EMACS <CR>
```

**NOTE:** You will be returned to the .YTEX file. Press M-< to bring the cursor to the beginning of the file. Then press ↑U. At the bottom of your screen you will see:

C-U

```
==>  1 3 ↑N
```

**NOTE:** Your cursor will be moved to the line containing the error. You can see that the command \bginnofill is spelled incorrectly. Add an e between b and g, and save the file:

```
==>  ↑X↑S
      ↑X↑Z
```

Written OZ:<[USERNAME]>LETTER.YTEX.2

@

```
==>  YTEX LETTER.YTEX <CR>
```

This is TeX, MIT Version 2.2 (preloaded format=ytex-cm 87.6.24)

```
(OZ:<[USERNAME]>LETTER.YTEX.2 { YTEX version 2.0} { No fixes} { \typesize=11.0pt}
(OZ:<LIZ>MACROS.YTEX.6) [1]
```

Output written on OZ:<[USERNAME]>LETTER.DVI.1 (1 page, 864 bytes).

Transcript written on OZ:<[USERNAME]>LETTER.LST.2.

**NOTE:** The input file is now error free and has been accepted by the YTeX program, which has produced as output a .DVI print file and a .LST transcript file, as you will notice when you do a directory listing:

Ⓞ

```
==> DIR <CR>
```

```
OZ:<[USERNAME]>
IDIOTS-GUIDE.YTEX.61,62,63
HELLO.TXT.1
LETTER.DVI.1
    .LST.2
    .YTEX.3
LOGIN.CMD.1
LOGOUT.CMD.1
MAIL.TXT.1,2
[USERNAME].BABYL.1
```

Total of 10 files

**NOTE:** Your formatted and printed output will look like the following page (you will learn how to print your document in the next section):

13 July 1987

Professor T. MacDuff  
Department of Computer Science  
University of Rochester  
Rochester, NY 14627

Dear Professor MacDuff:

I would be happy to have you visit our laboratory the next time you are in Boston. I will be out of town until *August 16*, but any time after that would be fine. Let me know when you have finalized your plans.

Yours sincerely,

Professor T. MacBeth  
MIT Artificial Intelligence Laboratory

TM:lh

## 6. PRINTING FILES

### 6.1 Printers Available at the A.I. Laboratory

There are a number of printers at the lab that can be accessed from OZ and used to print either "raw" text files or .DVI formatted files. There are three large, fast QMS 2400 printers that are capable of printing letter quality output on a variety of paper sizes. The fourth printer is a QMS PS-800 LaserWriter. The output quality of this printer is somewhat better, but it is much slower, and limited in what it can do. Two of the QMS 2400's and the QMS PS-800 are PostScript printers. The various printers, their names, and their locations are listed below:

7th Floor Laser Printer (PS-2400); Home: Xerox Room, 744; Name: PRAVDA

7th Floor LaserWriter (PS-800); Home: across hall from Room 707; Name: LE-MONDE

8th Floor Laser Printer (PS-2400); Home: Xerox Room, 800d; Name: DAILY-PLANET

9th Floor Laser Printer (2400); Home: near Machine Shop; Name: NATIONAL-ENQUIRER

### 6.2 Checking the Printer Status

To check the status of the printers, use the command `LPQ <CR>`. This command will list the status of all the printers in building NE43, including the A.I. Laboratory. To see the status of a particular printer, type `LPQ [PRINTER NAME] <CR>`. The status report will tell you whether the printer is currently working, what job it is now printing, and what other jobs are in the *queue* waiting to be printed. A typical status report will look like this:

The DAILY-PLANET queue:

8th Floor QMS is ready and printing

Time	Owner	Job Files	Size
*16:44	dejong	567 OZ:OZ:<DEJONG.THESIS>PRO.DVI.6	83688
16:48	liz	570 OZ:OZ:<POGGIO.DRAFTS>PIRELLI.DVI.3	2345
16:50	liz	571 OZ:OZ:<LIZ>REMIND.TXT.45	12765
16:55	little	573 Screen Hardcopy	8930

The most recent job printed was:

16:42 djr               PROJECT.DVI.4

The \* will be beside the job that is currently printing. If there are no jobs in progress or in the queue, you will see the following message:

```
The DAILY-PLANET queue:
8th Floor QMS status: idle
The queue is empty.  The most recent job printed was:
16:42 djr          PROJECT.DVI.4
```

## 6.3 Printing a File

### a. Defining a Printer

After you have checked the printer status, you should select the printer you wish to use. You may choose the printer with the shortest queue or the one closest to your work area, or you may choose LE-MONDE for higher quality (but slower) output. Once you have selected a printer, you must inform OZ of your choice. To do this, type the command `DEFINE PRINTER: [PRINTER NAME] <CR>`. Once you have entered this command, OZ will assume you are concerned only with the printer you have defined each time you send a job to print or enter the LPQ command. To change printers, use the `DEFINE` command again with a different printer name. If you usually use the same printer, you can define it in your `LOGIN.CMD` file. Ask an expert to show you how to do this.

### b. Sending a Job to the Printer

You can send a number of different types of jobs to the printer. You may send any file to the printer in its "raw" form. The output will look like it does on your terminal screen, including the YTeX commands. To print a file formatted by the YTeX program, send the .DVI file (described in Section 5) to the printer. If you are using a LISP Machine, you can get a printout of your screen display.

To print a file, use the command `LPR [FILE NAME] <CR>`. If you have defined your printer as shown above, your file will automatically be put in the print queue of the appropriate printer and will be printed in turn. Use the LPQ command to see what position your job is at in the queue.

### c. Stopping a Print Job

If for some reason you decide you do not wish to print a file that has been sent to the printer, you can remove a job from the waiting queue, or stop it if it has already started to print. When you use the LPQ command, you will notice that there is a number for each job listed before the file name in the Job column. To remove, or *dequeue*, a job, type the command `LPRM [JOB NUMBER] <CR>`. The command `LPRM [USERNAME] <CR>` will dequeue all jobs sent by the specified user.

## 6.4 Printing Errors

Sometimes when you attempt to check the status or send a job to a printer, there will be a pause and you will see the message: `Host is not up`. The *host* is a special computer that routes jobs from the different computers to the various printers. The host for most of the printers is named PREP. If PREP is down, you will not be able to print your job. Usually these outages are of short duration; wait a few minutes and try to print the file again.

Once you have successfully sent your job to the printer, other types of messages may appear on your screen if the printer gets an error while printing your file. These messages are of the following form:

```
[Message from daemon@prep.ai.mit.edu 16-Jul-87 15:59:09]
7th Floor LaserWriter needs paper.
```

These messages usually refer to a paper jam, an empty paper tray, or a similar physical problem. When this occurs, you will have to go to the printer to determine what is wrong and fix it. If the jobs in the queue seem to be taking an inordinately long time to print, it is a good idea to check the printer. Only the user whose job is currently printing will receive the error messages.

If the printer has no clear mechanical problems and still will not print, the printer, or its host, is probably "*wedged*". When this occurs, there is a pair of commands you may use to attempt to fix it. These commands will work only for the printer that is currently defined. Type `LPC ABORT <CR>` to stop any current processes, immediately followed by `LPC START <CR>` to resume. If the printer still does not work, consult an expert or log a service call for repair.

## 6.5 A Practice Session

In this Practice Session, you will print out the `EXAMPLE.TXT` file you worked on in Section 4, and the letter you created in the Section 5 Practice Session. Before beginning, you should locate the different printers so you will know where to retrieve your output.

0

```
==> LPQ PRAVDA <CR>
      LPQ LE-MONDE <CR>
      LPQ DAILY-PLANET <CR>
      LPQ NATIONAL-ENQUIRER <CR>
```

**NOTE:** Use these commands to check the status of the different printers. `DAILY-PLANET` will be used as an example:

DAILY-PLANET queue:

8th Floor QMS is ready and printing

Time	Owner	Job Files	Size
*16:44	dejong	567 OZ:OZ:<DEJONG.THESIS>PRO.DVI.6	83688

The most recent job printed was:

16:42 djr                   PROJECT.DVI.4

NOTE: You can see from the above status report that the printer is ready and printing, and has one job in progress. Select the printer you wish to use and inform OZ of your choice:

@

==> DEFINE PRINTER: DAILY-PLANET <CR>

NOTE: Print your EXAMPLE.TXT file in both "raw" and processed form so that you can compare how they look. First, the "raw", unprocessed file:

@

==> LPR EXAMPLE.TXT <CR>

@

==> LPQ <CR>

DAILY-PLANET queue:

8th Floor QMS is ready and printing

Time	Owner	Job Files	Size
*16:50	[username]	570 OZ:OZ:<[USERNAME]>EXAMPLE.TXT.4	22342

The most recent job printed was:

16:44 dejong                   OZ:OZ:<DEJONG.THESIS>PRO.DVI.6

NOTE: Your job is now being printed. Next, run the YTeX program on the file EXAMPLE.TXT to produce the formatted file EXAMPLE.DVI (see Section 5 on using YTeX), and print it also:

@

==> LPR EXAMPLE.DVI.2 <CR>

@

==> LPQ <CR>

DAILY-PLANET queue:

8th Floor QMS is ready and printing

Time	Owner	Job Files	Size
*16:57	[username]	575 OZ:OZ:<[USERNAME]>EXAMPLE.DVI.2	24116

The most recent job printed was:

16:50 [username] OZ:OZ:<[USERNAME]>EXAMPLE.TXT.4

NOTE: You can see that EXAMPLE.TXT is now finished, and EXAMPLE.DVI is currently printing. Assuming you ran the YTeX program on the LETTER.YTEX file in the Practice Session for Section 5, you can now print the formatted file LETTER.DVI:

⓪

==> LPR LETTER.DVI.1 <CR>

⓪

==> LPQ <CR>

DAILY-PLANET queue:

8th Floor QMS is ready and printing

Time	Owner	Job Files	Size
*16:57	[username]	575 OZ:OZ:<[USERNAME]>EXAMPLE.DVI.2	24116
17:02	[username]	578 OZ:OZ:<[USERNAME]>LETTER.DVI.1	3761

The most recent job printed was:

16:50 [username] OZ:OZ:<[USERNAME]>EXAMPLE.TXT.4

NOTE: Notice that EXAMPLE.DVI is still printing, and LETTER.DVI is in the queue and will be printed next.

When the printer is finished, go retrieve your output. Printouts will either be in the printer output tray or in the bins near the printer, filed alphabetically by username. If you do not like the way your output looks, make changes to your.YTEX file, run the YTeX program again, and print the new.DVI file that is produced.

When you are done printing, you will have finished the last Practice Session in the guide. Kill all your jobs and logout:

⓪

==> RESET \* <CR>

⓪

==> LOGOUT <CR>

[Host connection closed.]



## Appendix A

## COMMAND SUMMARY

The commands in this summary are arranged in the order that they appear in the manual. Page numbers indicate where the command is introduced and explained.

## Section 1 - Getting Started

## Key Commands (p. 4, 5)

**<control>** (↑) – (left of letter keys); hold down while you press letter or symbol keys.

**<escape>** – (upper left); press and release, then press letter key. Used in conjunction with **<shift>** key as a meta key on some keyboards.

**<meta>** (M-) – (location varies); press in conjunction with letter or symbol keys. Not available on some keyboards; substitute keys are used.

**<pause>** – (lower left); hold down while pressing letter or symbol keys. Used as a meta key on some keyboards.

**<CR>** – (large key at right of letter keys); carriage return; press to send commands or to move down a line.

## Accessing OZ and Logging In (p. 6)

**14↑Z** – Type at the **>** (concentrator) prompt to access OZ.

**14T** or **OT** – Type at the concentrator prompt to access OZ using the TELNET protocol (requires a password).

**14S** or **OS** – Type at the concentrator prompt to access OZ using the SUPDUP protocol (maybe used from a personal computer).

**T** – Select to access OZ from a Lisp Machine. Use the **<select>** and **<help>** keys to list available options.

**[USERNAME] <CR>** – Type at the first **@** prompt to log in if using **14↑Z**.

**[USERNAME] [PASSWORD] <CR>** – Type at the first **@** prompt if using the TELNET protocol.

**TAKE LOGIN.CMD <CR>** – Re-read and re-execute your LOGIN.CMD file while logged in; re-initialize environment.

### Fail Safe Key Commands (p. 7)

↑C – Exit the current program and return to the top level. Press twice if once does not work.

↑L – Clear and redraw the video screen.

↑U – Erase current command and return to @.

↑\k – Return to the initial > (concentrator) prompt.

### On-Line Help (p. 8)

HELP <CR> – List all topics for which help is available on-line.

HELP ? <CR> – Show an abbreviated list of topics.

HELP [TOPIC NAME] <CR> – Replace [TOPIC NAME] with a subject from one of the lists.

### Identifying Yourself and Other Users (p. 9)

WATSON <CR> or INQUIRE <CR> – Enter the WATSON program to enter or change personal information.

ALL <CR> – Request that WATSON ask about all entries.

DONE <CR> – End WATSON session and update personal information.

WHOIS <CR> – List information for all users currently logged onto OZ.

WHOIS [USERNAME] <CR> – List personal information for a specific user.

FINGER <CR> – List, without WATSON information, all users currently logged onto OZ.

FINGER [USERNAME] <CR> – Find a specific user and see what he is currently doing on OZ.

SYS <CR> – List information about users in an abbreviated form.

### Detaching and Attaching (p. 9)

DETACH <CR> – Detach your current job and return to the opening @ prompt.

ATTACH [USERNAME] <CR> – Re-attaches your job, retaining features of the environment.

<CR> – Type at the Attach your detached job? prompt to re-attach previous job.

### **Killing Jobs and Logging Out (p. 9)**

**INF FO <CR>** – List all the jobs you currently have in progress.

**RESET [JOB NAME] <CR>** – Kill a specific job.

**RESET . <CR>** – Kill the last job used.

**RESET \* <CR>** – Kill all of your currently active jobs.

**LOGOUT <CR>** or **KK <CR>** – Log off OZ; session completed.

## **Section 2 - Manipulating Files**

### **Listing Your Directory (p. 13, 16)**

**DIR <CR>** – Show a list of the files in the current directory.

**VDI <CR>** – Show a detailed listing of the files in the current directory.

**NDIR <CR>** – Show a listing of the directory indicating the status of off-line files.

**CD [DIRECTORY NAME] <CR>** – Change to another user's directory or a subdirectory.

### **Creating and Editing Files (p. 14)**

**EMACS <CR>** – Initiate an EMACS job, if none is currently active, or return to the last file edited.

**↑X↑V** – Visit a file; you will be asked for the name of the file to visit.

**↑X↑F** – Find a file; you will be asked for the name of the file to find.

**EMACS [FILENAME] <CR>** or **EDIT [FILENAME] <CR>** – Enter the editor to work on the specified file.

### **Saving and Exiting Files (p. 14)**

**↑X↑S** – Write (“Save”) the changes made to the file.

**↑X↑Z** or **↑Z↑Z** – Exit the editor and return to the top level.

**Deleting, Recovering, and Restoring Files (p. 15, 16)**

**DELETE [FILENAME] <CR>** – Remove the specified file.

**EXPUNGE <CR>** – Erase all deleted files from memory.

**UNDELETE [FILENAME] <CR>** – Recover a file that has been deleted but not yet expunged.

**GROVEL <CR>** – Enter the GROVEL program to recover from tape files that have been expunged.

**HELP GROVEL <CR>** – Get information about the GROVEL program.

**RETRIEVE [FILENAME] <CR>** – Restore files that have been archived on backup tapes.

**Copying Files (p. 16)**

**COPY [OLD FILENAME] [NEW FILENAME] <CR>** – Copy a file within the same directory.

**COPY <[DIRECTORY NAME1]>[OLD FILENAME] <[DIRECTORY NAME2]>[NEW FILENAME] <CR>**  
– Copy a file to or from another directory.

**Using Subdirectories (p. 16)**

**DIR <[HOME DIRECTORY].\*> <CR>** – List all subdirectories associated with a specific home directory.

**BUILD [SUBDIRECTORY NAME] <CR>** – Create a subdirectory associated with the current home directory.

**LIST <CR>** – List current attributes of directory or subdirectory.

**? <CR>** – List available options.

**MAX <ESC> [NUMBER] <CR>** – Set the limit of subdirectories allowed within the current directory.

**WORK [NUMBER] <CR>** – Set the working page quota.

**PERM [NUMBER] <CR>** – Set the permanent page quota.

**<CR>** – Exit BUILD program.

**Section 3 - Using Electronic Mail****Basic MM Commands (p. 22)**

**MM <CR>** – Enter the MM mail program.

**HELP <CR>** – From within MM, get instructions on how to use the program.

**EXIT <CR>** – Leave MM and return to the top level (does not kill the program).

**RESET MM <CR>** – Kill the MM program.

**Sending Mail in MM (p. 22)**

**SEND <CR>** – From the **MM>** prompt, enter the sending subprogram; from the **S>** prompt, send the message most recently entered.

**↑E** – Edit a message prior to sending.

**↑N**, then **Y** – Abort; cancel a message without sending.

**↑B** – Specify a file to be sent using **MM**.

**↑Z** – Complete a message and prepare to send.

**Reading Mail in MM (p. 23)**

**READ <CR>** – From the **MM>** prompt, enter the reading subprogram and read the first message.

**READ [NUMBER] <CR>** – Read only the message with the specified number.

**NOTE:** The following commands should be entered at the **R>** prompt:

**REPLY <CR>** – Send a return message in reply to the one most recently read.

**FORWARD [USERNAME] <CR>** – Forward a message you have received to another user.

**DELETE <CR>** – Delete the message most recently read.

**NEXT <CR>** or **<CR>** – Read the next message in the mail file.

**QUIT <CR>** – Return to the **MM>** prompt without reading all the messages in the mail file.

**BABYL Commands (p. 24)**

**BABYL <CR>** – Enter the **BABYL** program; when first used, create a **BABYL** file.

**Q** – Leave the **BABYL** program and return to top level.

**?** – Get help within **BABYL**; **\*** will give a list of all commands.

**↑X R** – Enter the **BABYL** program from within **EMACS**.

**Basic Editing Commands for BABYL (p. 25)**

**↑D** – Delete the character under the cursor.

**↑F** – Move the cursor Forward one space.

**↑B** – Move the cursor Backward one space.

**↑N** – Move the cursor to the Next line.

**↑P** – Move the cursor to the Previous line.

**Sending Mail with BABYL (p. 25)**

**M** – Initiate a message to send.

**↑Z↑Z** – Send the message just entered.

**M-X Abort** – Kill a BABYL message without sending.

**Reading Mail in BABYL (p. 26)**

**B** – List messages in [USERNAME].BABYL file

**D** – Delete the current message.

**E** – Expunge deleted messages from [USERNAME].BABYL file.

**F** – Enter editing mode.

**G** – Get any mail that has come in since you started theBABYL session.

**N** – Go to the Next message.

**O** – Put the message into another file; you will be prompted for file name.

**P** – Go to the Previous message.

**Q** – Quit the BABYL program.

**U** – Undelete the previous message.

**Mail Short Cuts (p. 28)**

**MAIL [USERNAME] <CR>** – Send mail from the top level to recipient's mail file.

**SEND [USERNAME] <CR>** – Send mail from the top level; will appear immediately on recipient's screen, and will not be put in mail file.

**HUH <CR>** or **WHAT [NUMBER] <CR>** – Look at the last SEND messages received. Using the [NUMBER] option allows you to see multiple past messages.

**SHOUT <CR>** – Send a one-line immediate message to all users.

**RPMAIL <CR>** – Read all your waiting mail messages at once; you will have the option of deleting or storing all the messages.

## Section 4 - Editing with EMACS

### On-Line Help and the TEACH-EMACS Tutorial (p. 33)

↑\_ - Type for help within EMACS. Enter ?<CR> at the prompt for more information.

↑X I - Enter a tree-structured information listing. Enter ↑X to exit.

TEACH-EMACS <CR> - Begin the TEACH-EMACS tutorial.

### Using Buffers (p. 33)

↑C↑B - List all currently existing buffers.

↑C B - Set up a prompt to choose which buffer to view or to use.

↑C K - Kill the specified buffer (default is the current buffer).

↑G - Abort the last command set in motion.

RESET EMACS <CR> - Kill all existing buffers (execute at top level).

### Entering EMACS (p. 34)

EMACS <CR> - Initiate an EMACS job, if none is currently active, or return to the last file edited.

↑X↑V - Visit a file; you will be asked for the name of the file to visit.

↑X↑F - Find a file; you will be asked for the name of the file to find.

EMACS [FILENAME] <CR> or EDIT [FILENAME] <CR> - Enter the editor to work on the specified file.

### Using Windows (p. 34)

↑C 2 - Open a second window.

↑C 0 - Move from current window to other window.

↑C 1 - Resume full-screen editing. Kill whichever window you are not currently in.

**Moving within the File (p. 35)**

↑V – Move down one screen of text.

M-v – Move up one screen of text.

M-< – Move to the beginning of the file.

M-> – Move to the end of the file.

↑P – Move to the Previous line of text.

↑N – Move to the Next line of text.

↑F – Move Forward one letter.

↑B – Move Backward one letter (same as <back space> key).

M-F – Move Forward one word.

M-B – Move Backward one word.

↑A – Move to the beginning of the current line.

↑E – Move to the End of the current line.

M-[ – Move to beginning of the current paragraph.

M-] – Move to end of the current paragraph.

↑U [LINE NUMBER] ↑N – Move to the specified line number.

**Inserting and Transposing Text (p. 36)**

↑O – Open a blank line at the cursor.

↑T – Transpose two letters.

M-T – Transpose two words.

**Deleting and Undeleting Text (p. 36)**

↑D – Delete the character under the cursor.

↑K – Delete (“Kill”) the current line from the cursor to the end.

M-D – Delete one word forward.

M-<del> – Delete one word back.

↑X <del> – Delete one sentence back.

↑Y – Reinstate (“Yank”) the text most recently deleted.



**Searching for Strings of Text (p. 36)**

↑S – Search forward for a string in the file.

↑R – Search backward for a string in the file.

<escape> – End the current search.

**Moving and Copying Blocks of Text (p. 37)**

M-@ or ↑<space bar> – Mark the beginning of a region of text; move the cursor to the end of desired region.

↑W – Delete (“Wipe”) the text within the marked region.

M-W – Add the marked text to the move buffer and also keep a copy at the original location

↑Y – Reinststate (“Yank”) the buffered text at a new (or the same) location.

**Special Character Commands (p. 37)**

M-L – Convert the current word to Lower case letters.

M-Q – Reformat the current paragraph.

M-S – Center the text preceding the cursor.

M-U – Convert the current word to Upper case letters.

M- \_ – Underline the word preceding the cursor.

M-\$ – Check the spelling of the current or previous word.

**Extended (M-X) Commands (p. 38)**

M-X APROPOS <CR> – Information about M-X commands.

M-X AUTO FILL <CR> – Switch on and off Auto Fill Mode.

M-X CHECK MAIL <CR> – List the mail currently waiting to be read.

M-X CORRECT SPELLING <CR> – Check and correct the spelling of the entire file.

M-X Dired <CR> – List the files in your directory from within EMACS.

M-X LISP MODE <CR> – Set the <tab> key to properly indent LISP programs.

M-X LOAD LIBRARY SORT <CR> – Load the sort routine library

**M-X QUERY REPLACE <CR>** – Replace specific occurrences of a string within a file. Type **<space bar>** to replace or **<delete>** to leave as is.

**M-X REAP <CR>** – Delete all but the two most recent versions of each file.

**M-X REPLACE STRING <CR>** – Replace all occurrences of a string within the file.

**M-X SORT LINES <CR>** – Alpha-numerically sort the file by lines

**M-X SORT PARAGRAPHS <CR>** – Sort the file by paragraphs

**M-X UNDO <CR>** – Reinstate the most recently killed material.

**↑G** – Abort an extended command without executing (use twice if necessary).

### **Using Keyboard Macros (p. 40)**

**↑X (** – Begin defining a keyboard macro.

**↑X )** – End the definition of the current keyboard macro.

**↑X E** – Execute the most recently defined macro (may be used repeatedly).

**↑U [NUMBER] ↑X E** – Execute the macro the specified number of times.

**↑G** – Abort the current macro

### **Saving Files and Exiting EMACS (p. 40)**

**↑X↑S** – Write (save) changes made to the current file.

**↑X↑W** – Write changes to a file specified by the user.

**↑X↑Z** or **↑Z↑Z** – Leave the editor and return to the top level.

**↑C** – Exit the editor without saving any changes.

**RESET EMACS <CR>** – Kill any EMACS jobs currently in progress.

## **Section 5 - Text Formatting with YTeX**

### **Running and Debugging YTeX (p. 51, 52)**

**YTEX [NAME-OF-FILE.YTEX] <CR>** – Run the YTeX program on the specified file.

**X <CR>** – Type at the ? prompt when YTeX is running to stop YTeX execution.

**<CR>** – Type at ? prompt to skip over an error and continue YTeX execution.

**↑U [LINE NUMBER] ↑N** – Use at the beginning of a .YTEX file to locate a specific line.

**↑C** – Emergency exit from the YTeX program.

**RESET YTEX <CR>** – Kill any currently active YTeX jobs.

## Section 6 - Printing Files

### Checking the Printer Status (p. 59)

LPQ <CR> – If printer is not defined, list status and queues of all printers in building NE43; if printer is defined, list status and queue of pre-defined printer.

LPQ [PRINTER NAME] <CR> – Check the status and queue of the specific printer.

### Defining a Printer (p. 60)

DEFINE PRINTER: [PRINTER NAME] <CR> – Set the printer you wish to use.

### Sending a Job to the Printer (p. 60)

LPR [FILENAME] <CR> – Send the specified file to the pre-defined printer.

### Stopping a Print Job (p. 60)

LPRM [JOB NUMBER] <CR> – Remove a job from the print queue of the pre-defined printer.

LRPM [USERNAME] <CR> – Dequeue all the jobs sent by the specified user.

### Unwedging the Printer (p. 61)

LPC ABORT <CR> – Stops processes on the currently defined printer.

LPC START <CR> – Re-starts processes on the currently defined printer.

## Appendix B

## GLOSSARY OF TERMS

The following is a glossary of terms used within this manual. Terms printed in *italics*, both throughout the guide and within definitions, are listed and defined in this section. Some of these terms have a specific meaning in “OZ terminology” which should not be confused with their more common usage.

**ACCOUNT** – A user’s means of access to OZ. An *account* provides you with a *username* (or *login name*), a *password*, and a certain amount of memory space. Some kinds of *accounts* give you greater privileges than others.

**AT-SIGN** – The `@` character. This is used as a *prompt*, and indicates that you are at the *exec* or *top level* on OZ. The `@` indicates that OZ is ready to receive input from the user. See also *prompt*.

**BACKSLASH COMMANDS** – Commands preceded by the character `\`. These are used in the *text formatting* programs YTeX and LaTeX to specify how the final document should look.

**BUFFER** – An unsaved *file* which holds work currently in progress. See also *file*, *minibuffer*.

**CHAOSNET** – A network system that allows communication between MIT computers, including most of those at the A.I. Laboratory.

**CHARACTER COMMANDS** – Commands that are entered by typing one of the special keys *control* or *meta* along with a letter or symbol. These are used to perform special operation within the EMACS program.

**COMMAND** – An instruction which is given to OZ or to one of the programs it is running, to indicate what the computer should do.

**COMMENTS** – Strings of text entered as documentation, or as notes to the programmer or user. These are not interpreted by the computer. *Comments* are preceded by the character `%`.

**CONCENTRATOR** – A router that connects the various terminals to the various computers in the A.I. Laboratory. The *concentrator* allows users to access more than one computer.

**CONTROL KEY** – A special key that gives alternate meanings to the letter and symbol keys. The *control key* is usually to the left of the letter keys on the keyboard, and is used in many of OZ's operations and programs. Represented in this manual by the character ↑. Also see *meta key*, *escape key*.

**CRASH** – Slang term for a computer malfunction. When a *crash* occurs, the computer will be *down*, and all work must be postponed.

**CURSOR** – The small rectangle of light on the screen that indicates where the computer is currently directing its “attention”. The *cursor* may be moved around in order to enter, remove, or manipulate characters and text.

**DEQUEUE** – To remove a file from the *queue* in which it is currently waiting.

**DIRECTORY** – A collection of *files* belonging to a specific user. Each user has a *home* or *root directory* and possibly some associated *subdirectories*. The system of *directories* is the way the computer organizes the files belonging to various users.

**DISPLAY STYLE** – See *vertical mode*.

**DOCUMENTATION** – Explanations of how *programs* work and how to use them. *Documentation* may be either *on-line* or contained in printed manuals.

**DOWN** – Adjective meaning “not working.” The computer may be *down* due to a malfunction or for periodic maintenance. See also *crash*.

**.DVI FILE** – A file produced by the YTeX program. This file contains the instructions that will cause the printer to output the final document in the style you have specified.

**EDITOR** – A *program* that allows the user to enter, remove, and manipulate *files* of text. See also EMACS.

**EMACS** – The *editor* used on OZ and other A.I. Laboratory computers.

**ESCAPE KEY** – A key used to perform special functions on OZ. The *escape key* can be used for file or command name completion, and on some keyboards, it is used along with the <shift> key to represent the *meta key*. Also see *control key*, *meta key*.

**EXEC** – The highest level of the *operating system*, from which *programs* are initiated. Also known loosely as *top level*.

**EXTENDED COMMANDS** – Commands used in EMACS to perform special operations or functions. These commands are initiated by typing M-X followed by the command word(s), and are also known as *M-X commands*.

**FILE** – A unit of stored information. Files may be “raw” (containing only text), or they may be “processed” (run through a *text formatting* program). See also *directory*, *buffer*.

**FORK** – OZ terminology for *program* or *job*. For example, if you have an EMACS job running, you are said to have an EMACS *fork*.

**HOME DIRECTORY** – The main *directory* of a user’s *files*. This usually has the same name as the person’s *username*, and is allocated when an *account* is set up. Also known as *root directory*. Contrast with *subdirectory*.

**HORIZONTAL MODE** – One of the two equation formatting modes in the YTeX program. Equations in *horizontal mode* are typed and appear within a line of text. This is also known as *text style*. Contrast with *vertical mode*.

**HOST** – A computer that allows another device to run. For example, the computer PREP is a *host* for the printers, and manages their flow of work.

**JOB** – See *program* and *fork*.

**KEYBOARD MACRO** – A function defined by the user that allows a sequence of commands to be executed by pressing a single character command.

**KILL** – To stop a particular *program*. A *program* should be killed when the user does not wish to use it any longer.

**LaTeX** – One of the *text formatting* programs available on OZ. See also YTeX.

**LOGIN NAME** – See *username*.

**.LST FILE** – A file produced by the YTeX program. This file contains a transcript of your YTeX session, including error messages.

**MACRO** – A special function or operation that is defined by the user. *Macros* can be used in a variety of *programs*, and may be organized into *files* for regular use.

**META KEY** – Another special key used to change the meaning of letter and symbol keys. This is used especially within the EMACS program. The *meta key* may be represented by the <pause> key, or by a combination of the *escape key* and <shift> key, depending on your keyboard. Represented in this manual by the letter M-. See also *control key*, *escape key*.

**MIGRATED FILE** – See *off-line*.

**MINIBUFFER** – An auxiliary *buffer* used for entering commands while in EMACS. See also *buffer*.

**M-X COMMANDS** – See *extended commands*.

**OFF-LINE** – Adjective referring to files that have been *migrated*, or taken off OZ and archived on backup tapes. This is done to save space in OZ's memory. Archived files can be restored when needed.

**ON-LINE** – An adjective meaning, loosely, “on the computer”. *Documentation* is said to be *on-line* if you can access it from your terminal.

**OPEN TABLE** – In YTeX, a table that extends over one or more page breaks.

**OPERATING SYSTEM** – The main *program*, or group of programs, which runs the computer, and which enables many people to use the computer at once to run a variety of subsidiary programs. The *operating system* on OZ is called TWENEX, or TOPS-20.

**OZ** – The name of the computer used throughout this guide. OZ is a DEC PDP-10 computer located on the ninth floor of building NE43.

**PASSWORD** – A name or string of characters chosen by the user which verifies the user's identity to the computer. This is typed after the *username* when logging in.

**PROGRAM** – In general, a series of instructions to be performed by the computer. The term may refer to instructions written by a user in a computer language such as LISP, or to the powerful complex tools, such as EMACS and YTeX, which can be called upon to perform specific tasks. See also *fork* and *job*.

**PROMPT** – The symbol which appears at the left edge of your screen when the computer is waiting for user input. Different *programs* have different *prompts*. The *exec* prompt on OZ is the *at-sign*, or @.

**QUEUE** – A line in which *files* are waiting and advancing one at a time to be processed. For example, the print *queue* contains files waiting to be printed. Also used as a verb meaning “to put into the queue”. See also *dequeue*.

**REGIONS** – In EMACS, blocks or sections of text that are marked for moving, changing, or deleting.

**ROOT DIRECTORY** – See *home directory*.

**STATUS LINE** – In EMACS, the line at the bottom of the screen that tells you what you are currently working on and the current status of your file.

**STRING** – A group of letters, numbers, or symbols of arbitrary length.

**SUBDIRECTORY** – A directory that is created from the main, or *home directory*. *Sub-directories* are associated with a user's *home directory*, and are useful for organizing a large number of *files*.

**TEXT FORMATTING** – A method of processing text to achieve professional looking output. The *text formatting programs* Y $\TeX$  and L $\TeX$  operate on files containing *backslash commands*. They process these files and return new files containing instructions to be interpreted by the printer.

**TEXT STYLE** – See *horizontal mode*

**TYPESETTING** – See *text formatting*.

**TOP LEVEL** – Where you are when you are logged in but have not yet initiated any programs. See also *exec*.

**UP** – Adjective meaning that a computer, printer, etc. is operational. Contrast with *down*.

**USERNAME** – A name that uniquely identifies a user to the computer. The *username* is used to send and receive mail, and identifies the user's *files* and *jobs*. Also known as *login name*.

**VERSION** – The number assigned by the computer to a *file* indicating how many times that file has been stored. It does not mean that that many *versions* currently exist, since the earlier ones will have been deleted.

**VERTICAL MODE** – One of the equation formatting modes in Y $\TeX$ . Equations in *vertical mode* are set off from the text and may extend over several lines. This is also known as *display style*. Contrast with *horizontal mode*.

**VISITING** – In EMACS, entering a file either to look at it or to make changes to it.

**WEDGED** – A slang term used to describe a machine that is down, especially one that remains non-operational despite routine attempts to fix it.



**WILDCARD** – A character used to represent a variety of possibilities. This can be used, for example, to specify a number of file names that all have some word in common. The *wildcard* is represented by the character \*, and may be used within a variety of different programs.

**WINDOWS** – A feature that allows you to view and work on more than one file, process, etc. simultaneously.

**YTeX** – The *text formatting* program explained in this manual. This program takes a .YTEX file containing *backslash commands*, processes it, and returns a .DVI file that contains instructions for the printer. See also LaTeX.

## Appendix C

**OTHER RESOURCES****Sections 1 and 2**

**TOPS-20 User's Guide**, Digital Equipment Corporation, Marlboro, MA, 1980. (Available by mail from DEC).

**Section 3**

Stallman, Richard M., **Lisp Machine ZMail Manual**, A.I. Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1983. (Available from the A.I. Laboratory publications office).

**Section 4**

Stallman, Richard M., **EMACS Manual for TWENEX Users**, A.I. Laboratory Memo 555, Massachusetts Institute of Technology, Cambridge, MA, 1983. (Available from the A.I. Laboratory publications office).

Stallman, Richard M., **EMACS: The Extensible, Customizable, Self-Documenting Display Editor**, A.I. Laboratory Memo 519A, Massachusetts Institute of Technology, Cambridge, MA, 1981. (Available from the A.I. Laboratory publications office).

**Section 5**

Brotsky, Daniel, **How to Use YTeX**, A.I. Laboratory Working Paper 273, Massachusetts Institute of Technology, Cambridge, MA, 1985 and 1986. (Available from the A.I. Laboratory publications office).

Knuth, Donald E., **The TeXbook**, Addison-Wesley Publishers, Reading, MA, 1984 and 1986. (Available through the MIT Coop).

Lamport, Leslie, **LaTeX Document Preparation System: User's Guide and Reference Manual**, Addison-Wesley Publishers, Reading, MA, 1986. (Available through the MIT Coop).

Lamport, Leslie, **Using LaTeX on MIT-OZ, Version 2.08**, Massachusetts Institute of Technology, Cambridge, MA, 1985. (Available from the A.I. Laboratory publications office).

Samuel, Arthur L., **First Grade TeX: A Beginner's Manual**, Report STAN-CS-83-985 (Version 1), Stanford University Department of Computer Science, Stanford, CA, 1983. (Available by mail from Stanford University).

**Using TeX on MIT-OZ**, A.I. Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1985. (Available from the A.I. Laboratory publications office).

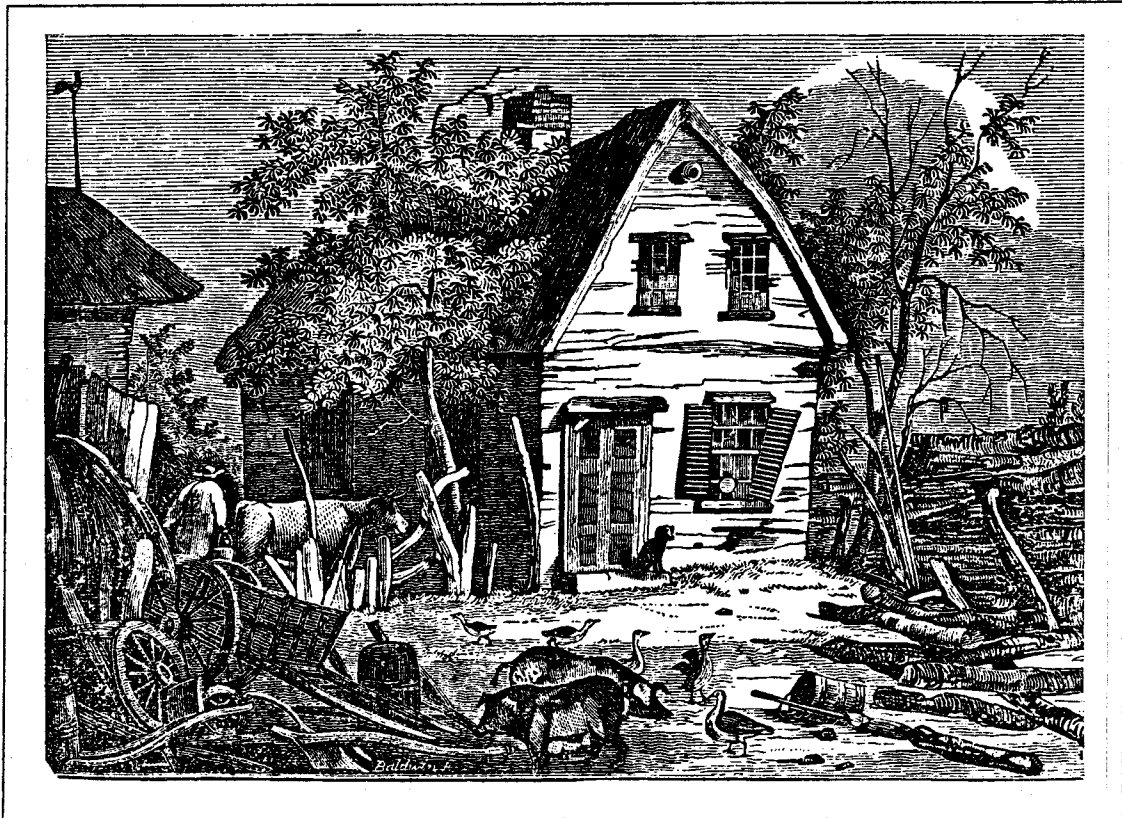
## Section 6

Jones, Scott A., **Printer Cheat Sheet**, available for all A.I. Laboratory printers, Massachusetts Institute of Technology, 1986. (Posted copy located on or near each printer; also available from the A.I. Laboratory publications office).

McCall, Bede B., **MacDoc: An Introduction to Computational Resources at MIT LCS and AI Lab**, Version 2.0, Massachusetts Institute of Technology, 1986. (Available from the Laboratory for Computer Science publications office).

# IDIOT'S GUIDE TO OZ

A MANUAL FOR THE COMPLETE BEGINNER,  
INTRODUCING  
*EMACS*, THE WORD PROCESSING PROGRAM,  
AND  
*TEX*, THE TEXT FORMATTING PROGRAM



William Gilson