Answers to Even-numbered Exercises 5

2. How can you keep other users from using write to communicate with you? Why would you want to?

   Give the command **mesg n** to keep other ordinary users from writing to your terminal. You can take this action when you do not want to be disturbed or are viewing something on your screen that you do not want overwritten.

4. List some differences between talk and write. Why are three different communications utilities (talk, write, and an email client) useful? Describe a situation in which it makes sense to use

   a. A mail program instead of talk or write

   b. talk instead of write

   c. write instead of talk

   The write utility

   a. Displays a line on the other user's screen only after you press RETURN.

   b. Does not work over a network.

   c. Mixes messages from you and the person you are communicating with.

The talk utility

a. Displays each character as you type it.

b. Works over a network.

c. Separates messages from you and the person you are communicating with.

You can always send email regardless of whether the user receiving the email is logged in and regardless of whether the user logs in on the same machine as you do.

In order to use talk, the user you want to communicate with must be logged in on a machine connected to a network you are connected to and must be accepting messages (mesg).

In order to use write, the user you want to communicate with must be logged in on the same machine as you and accepting messages (mesg).

6. How can you find out which utilities are available on your system for editing files? What utilities are there for editing on your system?

   Give the command **apropos editor**. Most systems have vim, ex, ed, and more.

8. What happens when you use diff to compare two binary files that are not identical? (You can use gzip to create the binary files.) Explain why the diff output for binary files is different from the diff output for ASCII files.

   When you compare binary files with diff, diff displays a message saying the files differ when the files differ or no message when the files are the same. The diff utility compares ASCII files on a line-by-line basis; it is not designed to compare binary files on a byte-by-byte basis. Use cmp to compare binary files in that manner.

10. What is the result of giving the which utility the name of a command that resides in a directory that is *not* in your search path?

    The which utility displays a message saying that the command you are looking for is not in the list of directories that are in your search path. For example,

```
$ which me
/usr/bin/which: no me in (/usr/local/bin:/bin:/usr/bin:/usr/X11R6bin:.
```

12. Experiment by calling the file utility with names of files in **/usr/bin**. How many different types of files are there?

    Approximately twenty, not counting files that you cannot read as a nonprivileged user.

14. Recreate the **colors.1** and **colors.2** files used in Figure 5-8 on page 127. Test your files by running **diff –u** on them, and see whether you get the same results as in the figure.

```
$ cat colors.1
red
green
yellow
pink
purple
orange

$ cat colors.2
red
blue
green
yellow
orange.
```

16. Repeat exercise 7 using the file **phone.gz**, a compressed version of the list of names and phone numbers. Try to consider more than one approach to each question, and explain how you chose your answer.

    You can either decompress the file using gunzip, giving the same commands as exercise 7 used once the file is decompressed, or use zcat and a pipe to display the results without creating an intermediate file as shown following:

```
  $ zcat phone.gz | grep "Ace Electronics"
  $ zcat phone.gz | sort
  $ zcat phone.gz | uniq
```

    Which technique you use makes a significant difference only if the **phone.gz** file is large, in which case it is an issue of what you are doing and a tradeoff between using more CPU (processor) time and less hard disk storage or vice versa.

    When you are giving one of the commands one time only, using a pipe is more efficient. When you want to give more than one of the commands or want to give one of the commands repeatedly, it is more efficient to decompress the file one time using gunzip and then process it repeatedly with grep, sort, or uniq, assuming you have sufficient disk space. Finally, the most inefficient technique as far as disk space goes and the most efficient as far as CPU (and your) time goes is to put the output of grep, sort, or uniq in new files using a redirect output symbol as shown following:

```
  $ zcat phone.gz | sort > phone.sort
```

18. Some mailers, particularly older ones, are not able to handle binary files. Suppose that you are mailing someone a file that has been compressed with gzip, which produces a binary file, and you do not know what mailer the recipient is using. Refer to the man page on uuencode, which converts a binary file to ASCII. Learn about the utility and how to use it.

    a. Convert a compressed file to ASCII, using uuencode. Is the encoded file bigger or smaller than the compressed file? Explain.

    If uuencode is not on your system, you can download it from **rpmfind.net**; it is part of the GNU **sharutils** package. Refer to "Installing, Upgrading, and Removing Packages" on page 457.

    The following command converts the file **memo.gz** to ASCII using uuencode. The **.uuencode** filename extension is not required.

    ```
    $ uuencode memo.gz > memo.gz.uuencode
    ```

    The resulting ASCII file is larger than the original binary file because uuencode includes control information.

    b. Would it ever make sense to use uuencode on a file before compressing it? Explain.

    No purpose is served by using uuencode to convert a binary file to ASCII before compressing it because compressing an ASCII file creates a binary file.