

5

ANSWERS TO EVEN-NUMBERED EXERCISES

2. What does the shell ordinarily do while a command is executing? What should you do if you do not want to wait for a command to finish before running another command?

The shell sleeps while a command is executing in the foreground. When you want to keep working while a command is running, run a command in the background by ending the command line with an ampersand (&).

4. What is a PID number? Why are these numbers useful when you run processes in the background? Which utility displays the PID numbers of the commands you are running?

PID stands for *process identification*. A PID number uniquely identifies the process running a command. When you run a command in the background, you can use its PID number as an argument to kill to stop the command from running. The ps utility displays PID numbers.

6. Refer to the info or man pages to determine which command will
- a. Display the number of lines in its standard input that contain the *word a* or *A*.

```
$ command | grep -wci a
```

- b. Display only the names of the files in the working directory that contain the pattern *\$(*.

```
$ ls *$(*)
```

- c. List the files in the working directory in reverse alphabetical order.

```
$ ls -r
```

- d. Send a list of files in the working directory to the printer, sorted by size.

```
$ ls -S | lpr
```

8. The `lpr` and `sort` utilities accept input either from a file named on the command line or from standard input.
- Name two other utilities that function in a similar manner.
`cat`, `grep`, and `uniq`
 - Name a utility that accepts its input only from standard input.
`tr` and `xargs`

10. Explain the following error message. Which filenames would a subsequent `ls` command display?

```
$ ls
abc abd abe abf abg abh
$ rm abc ab*
rm: cannot remove 'abc': No such file or directory
```

The shell expands the asterisk wildcard character before it passes a list of filenames to `rm`. As a result `rm` receives a list of files that includes `abc` twice. After `rm` removes `abc`, it generates an error message when it is asked to remove `abc` again. After giving the preceding `rm` command, `ls` does not list any files.

12. In experimenting with variables, Max accidentally deletes his `PATH` variable. He decides he does not need the `PATH` variable. Discuss some of the problems he could soon encounter and explain the reasons for these problems. How could he *easily* return `PATH` to its original value?

Because the shell has no way to find them, no commands or utilities, except builtins, will work.

```
$ ls
bash: ls: No such file or directory
```

The inability to run commands without specifying their pathnames makes the shell more difficult to use. Max can attempt to locate the `ls` command using `whereis`.

```
$ whereis ls
bash: whereis: No such file or directory
```

A simple way to return `PATH` to its original value is to log out and then log back in.

14. If you accidentally create a filename that contains a nonprinting character, such as a `CONTROL` character, how can you remove the file?

Use some unique characteristic of the filename along with wildcard characters to specify the file. Confirm the ambiguous file reference works by using `echo` before you attempt to use it with `rm`. For example, assume you want to remove the file named `abCONTROL-Txyz`. The following commands test an

ambiguous file reference to make sure it does not refer to any other files and then remove the file using the reference:

```
$ echo ab?xyz
abxyz
$ rm ab?xyz
```

16. Why do command names and filenames usually not have embedded SPACES? How would you create a filename containing a SPACE? How would you remove it? (This is a thought exercise, not recommended practice. If you want to experiment, create a file and work in a directory that contains only your experimental file.)

From a command line, when you want to refer to a file whose name contains an embedded SPACE, you must quote the SPACE. A SPACE is a special character to the shell; it typically separates tokens or words on the command line. Because it is tedious to refer to this type of file, most filenames do not include SPACES.

You can remove a file whose name contains a SPACE by quoting the SPACE:

```
$ rm dumb\ filename
or
$ rm "dumb filename"
```