Lab Exercise #1 -- The CSE UNIX Environment

This assignment familiarizes you with the computing environment that will be used for this course. You are to complete the experiments below and record the requested information on your worksheet.

A. CSE UNIX Accounts

At the "login" prompt, enter your username and password. If you don't know how to access the system, review the document "Introduction to the CSE Instructional System" (available as "~cse410/General/intro.system").

When you are ready to begin the experiments, move the mouse arrow into your working window (usually the lower leftmost window) so that you can enter commands at the command-line prompt.

B. The UNIX Operating System

The heart of the UNIX operating system is a set of system calls that are used to manage the system's resources (primarily the CPU, memory, and the file system). For example, the following system call could be used inside a C/C++ program to create a file named "/user/cse410/Labs/lab01.example":

```
creat( "/user/cse410/Labs/lab01.example", S_IRUSR );
```

A list of the system calls available on the CSE UNIX systems is available as:

```
~cse410/Labs/lab01.system_calls
```

An overview of the system calls is available through the "man" utility:

```
man -s 2 intro
```

Information about individual system calls is also available. For example:

```
man -s 2 creat
```

Users interact with UNIX by executing programs which invoke the appropriate system call(s) to perform the desired task(s).

Fortunately, the CSE UNIX environment already contains many programs written by system programmers to perform common tasks, such as GUIs (graphical user interfaces), shells (interactive command interpreters), and utility programs ("ls", "rm", etc).

In addition, most programming languages provide libraries of functions which make it easier for users to invoke system calls correctly. For example, the standard C library contains a function "fopen" to create and open files. Of course, "fopen" invokes "creat" to do some of the work.

Thus, most programmers employ a combination of a GUI, a shell, and their own programs containing library functions and system calls to perform tasks on a UNIX platform.

In this course, we will use UNIX as a case study. Thus, you will learn how UNIX implements modern operating system concepts, as well as how to be an effective user and programmer in a UNIX environment.

C. Shell Basics

When you access a Sun workstation in one of the CSE labs, a GUI (usually OpenWindows) begins executing and a shell (command interpreter) begins executing inside each window. Most students use the C shell command interpreter ("csh" or "tcsh"), although almost all shells are similar.

The C shell is a utility program designed to make it easier for users to perform common tasks. It repeatedly displays a prompt and waits for the user to enter input at the keyboard, interprets the input, and attempts to perform the task specified by the input.

1) Use "echo $shell" to display the name of the shell you are executing.

_____

The default shell prompt includes three pieces of information: the sequence number of the prompt, the name of the workstation you are using, and the name of the directory in which you are currently working. Note: the symbol "~" is used to represent the name of your home directory.

2) Give the shell prompt currently displayed in your working window.

_____

3) Describe the information displayed in that shell prompt.

_____

_____

_____

The shell accepts input from the keyboard and breaks up the line into "tokens" (individual words), using blanks and tabs as delimiters. Certain characters have special meanings. From the "man" entry for "csh":

> The characters &, |, ;, <, >, (, and) form separate words; if paired, the pairs form single words. These shell metacharacters can be made part of other words and their special meaning can be suppressed by preceding them with a '\' (backslash). A newline preceded by a '\' is equivalent to a space.

A simple command is a sequence of words; multiple simple commands can be entered on the same line if they are separated by semicolons.

4) Describe the output from "echo $shell; echo $shell".

_____

5) What happens if you enter a backslash and a new line after the semicolon?

_____

_____

The first word in a simple command is assumed to be the name of the built-in shell command or program that you wish to execute. If that word is not the name of a built-in shell command, the shell examines your search path and tries to find the indicated program.

From the "man" entry for "csh":

> If the command is a C shell built-in command, the shell executes it directly. Otherwise, the shell searches for a file by that name with execute access. If the command name contains a /, the shell takes it as a pathname, and searches for it. If the command name does not contain a /, the shell attempts to resolve it to a pathname, searching each directory in the "path" variable for the command.

For example, "cd" is a C shell built-in command to move to a different directory, and "/usr/bin/ls" is the name of a utility program to display the names of the files in a directory.

6) Use "echo $path" to display the directories in your search path.

_____

_____

_____

7) Briefly describe the effect of entering "/usr/bin/ls" at the prompt.

_____

8) Briefly describe the effect of entering "ls" at the prompt.

_____

D. UNIX File System Basics

The UNIX file system is organized as a tree of directories and files, with the directory "/" at the root of the tree. When you access your account, you are initially positioned in your home directory.

1) Use the command "pwd" ("print working directory") to display the absolute pathname of your home directory.

_____

2) Use the command "cd /user/cse410" to position yourself in the directory for this class, which is owned by the instructors and is used to store course files, then use "pwd" to display the absolute pathname of that directory.

_____

The shell recognizes several shortcuts related to home directories. The command "cd" (with no other tokens) changes directories to your home directory. When the character ’~’ is the first character in a token, the shell attempts to convert that token into the home directory of a user. As a special case, ’~’ by itself refers to the user’s home directory.

3) Briefly describe the effects of the following command sequence:

```
cd ~; pwd; \
cd ~cse410; pwd
```

_____

_____

The shell recognizes absolute pathnames (which begin with '/') and relative pathnames (which are assumed to be names relative to the current directory). In addition, the symbol "." is used to represent the current directory, and the symbol ".." is used to represent the parent of the current directory.

4) Give a sequence of commands to move into the home directory for "cse410", then into the "Labs" subdirectory and display its absolute pathname.

_____

5) Give a sequence of command to move from the "Labs" subdirectory to the "General" subdirectory and display its absolute pathname.

_____

6) The "ls" utility program displays the filenames (initial component of relative pathnames) of the files in the current directory. Use "man ls" to give a brief description of each of the following variations on "ls":

"ls": _____

_____

"ls -a": _____

_____

"ls -l": _____

_____

The shell recognizes wildcards when you specify filenames: '*' denotes a sequence of zero or more characters, and '?' denotes a single character.

7) Move into the "General" subdirectory of the "cse410" home directory, then list the filenames displayed by the following commands:

"ls in*":

_____

"ls intro*":

_____

"ls *e":

_____

"ls *e*":

_____

"ls *?e?*":

_____

"ls *.e*":

_____

E. Utility Programs

The CSE UNIX environment contains a wealth of utility programs, including some that are used to manage files, such as "cp", "mv", "rm", "mkdir" and "rmdir". Others are used to display the contents of text files ("cat", "more"), to edit text files ("vi", "emacs"), or to filter text files ("head", "cut", "grep").

1) The file named "~cse410/Labs/lab01.faculty_list" contains a list of faculty members from Computer Science and Engineering. Copy that file into your account as the file named "flist", then answer the questions below.

Command to copy the file: _____

Command to display contents of "flist": _____

2) For each of the commands given below, describe (in general terms) the effect of executing the command.

head flist: _____

head -8 flist: _____

tail flist: _____

tail -5 flist: _____

cut -c1-22,33-40 flist: _____

sort flist: _____

sort +3b flist: _____

sort +3br flist: _____

grep 'Ch' flist: _____

grep 'E' flist: _____

grep '^E' flist: _____

grep '7$' flist: _____

grep -n 'EB' flist: _____

grep -v 'EB' flist: _____

3) Give the command which will display all of the lines in "flist", sorted into order by office address (building, then room number).

_____

4) Give the command which will display the name and office address of each faculty member, but not their phone number.

_____

5) Give the command which will display all of the lines in "flist" which contain phone numbers that do NOT begin with "353".

_____

F. I/O Redirection and Pipelines

In a UNIX environment, programs have three standard I/O streams. By default, programs read from the standard input stream (using "stdin" in C) and write to the standard output and error streams (using "stdout" and "stderr" in C). The C shell provides mechanisms to redirect the standard I/O streams to files and to create pipelines between processes.

The characters '<', '>', and '>>' indicate that the next token is the name of a file to which the program's standard input or standard output is to be redirected. The character '&' indicates that the program's standard output and error streams should both be redirected to the same file (for example, "a.out >& temporary").

1) Consider the following sequence of commands:

```
echo 'Offices in PSS' > psslist
grep 'PSS' flist >> psslist
```

What is the overall effect of executing this sequence of commands?

_____

_____

Meaning of the symbol ">": _____

Meaning of the symbol ">>": _____

Command to display contents of "psslist": _____

The C shell also provides a mechanism to connect the standard output stream of one program to the standard input stream of another program, thereby creating a pipeline. From the "man" entry for "csh":

> A simple command, or a set of simple commands separated by | or |& characters, forms a pipeline. With |, the standard output of the preceding command is redirected to the standard input of the command that follows. With |&, both the standard error and the standard output are redirected through the pipeline.

The two mechanisms can be combined. For example, a pipeline might contain six different utility programs (each of which might have command-line options): the output of the first program is connected to the input of the second program, and so on. The first program might have its input redirect from a file, and the last program might have its output redirected to a file.

2) Consider the following sequence of commands:

```
echo 'Offices in EB' > eblist
grep 'EB' flist | sort +1b -2 >> eblist
```

What is the overall effect of executing this sequence of commands?

_____

_____

Meaning of the symbol "|": _____

Command to display contents of "eblist": _____