**Chapter 5**

**Path Names**

**1. Relative a nd Absolute Pathnames**

**1.1 Relative Pathnames**

The use of the ".." notation allows us to navigate the directory tree structure. The ".." symbol means "parent directory." Names with ".." in them are relative names because their meaning depends on where they are issued (the present working directory). we can string together several ".." symbols, separated by the / symbol and other directory names, to change directories. For example, if we are in portfolio and want to change to mary, we can do this with a cd command followed by the relative pathname between portfolio and mary like this (first using pwd to show where we are):

**$ pwd**

**/users/john/portfolio**

**$cd ../../mary**

**$pwd**

**/users/mary**

**$**

Directory or file references starting with .. are relative pathnames.

Directory or file references starting with a directory name are also relative pathnames. For example, if we are in the users directory, the directory reference john/portfolio is a relative pathname:

**$ pwd**

**/users**

**$cd john/portfolio**

**$pwd**

**/users/john/portfolio**

**$**

**1.2 Absolute Pathnames**

If we string together the unique name of all the intervening subdirectories in the file system to a particular subdirectory, we have created the absolute pathname for that directory. The absolute pathname allows us to switch to a directory no matter what my present working directory is. Absolute pathnames always start with a "/". we can navigate the file system by using absolute pathnames. So we could do something like this:

**$ pwd**

**/users/john**

**$ cd /users/mary**

**$ pwd**

**/users/mary**

**$ cd /tmp**

**$ pwd**

**/tmp**

**$ cd /users/john/portfolio**

**$ pwd**

**/users/john/portfolio**

**$**

Every directory or file on the file system has a unique absolute pathname. Although john may create a file called "test.txt" in his home directory and mary may create a file called test.txt in her home directory, the absolute pathnames of these files are different. John's is called /users/john/test.txt, and Mary's is /users/mary/test.txt.

**2. Directory Abbreviations**

|  |  |
| --- | --- |
| **Keyword** | **Description** |
| ~ | Your home (login) directory |
| ~username | Another user's home directory |
| . | Working (current) directory |
| .. | Parent of working directory |
| ../.. | Parent of parent directory |

**Examples**

**cp foo bar**

Copy a file named "foo" (in the current directory); name the copy "bar"

**cp foo ~/Documents**

Copy a file named "foo" (in the current directory) into your Documents directory

**cp foo ~/Documents/bar**

Copy a file named "a" (in the current directory) into your Documents directory and name the copy "bar"

**cp \*.jpg ~/Documents**

Copy all files with names ending in ".jpg" into your Documents directory

**cp -R Documents "Documents backup"**

Copy an entire directory named "Documents"; name the copy "Documents backup". The quotes are needed because of the space in the directory name.

**sudo cp -Rp /Users "/Users backup"**

Copy the entire /Users directory (including all of the user home folders inside it), preserving as much as possible of the files' information (ownership, permissions, etc, but not resource forks) as cp knows how to; name the copy "Users backup". Root access is required to use -p, so the example uses sudo to get root access temporarily.