

Some Unix Help

What is Unix?

Unix is an operating system.

An operating system is software that runs a computer. Be they personal computers, network workstations, network servers or mainframes, all computers have operating systems. The operating system of a computer provides the environment that is used to run applications software, such as a word processor or an Internet browser. In addition, the operating system provides the means for multiple computers to be linked together into a computer network. The operating system that most students have experience with before taking a first class in Computer Science one of the various versions of Microsoft Windows.

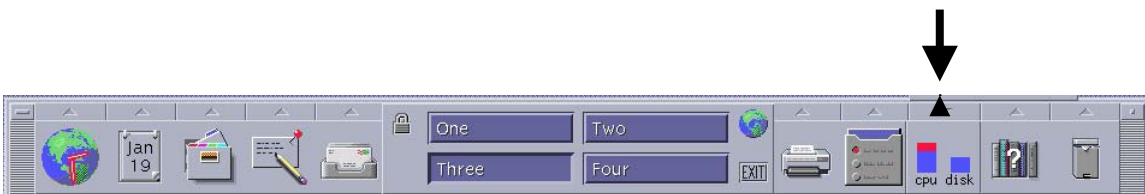
Unix was originally developed starting in 1969 at Bell Labs by Ken Thompson, Dennis Ritchie and many others. It has gone through extensive development to reach the OS that we see to day in the UCS and SL labs, etc., complete with a graphical user interface. One more recent branch of the Unix family is called Linux, which is designed to work on PCs. Some makers of the various flavors of Unix make Intel compatible versions. For example, the maker of the Solaris operating system, Sun Microsystems, offers a version of the latest Solaris that runs on PCs.

While it is to a certain extent true that all operating systems perform the same functions, they don't all do every function equally well. Unix is a remarkably stable and flexible operating system. Although still not commonly found on computers in home PCs or on office desktops, the power and stability of Unix makes it ideal for providing a substantial portion of the network infrastructure for businesses, universities, government and the Internet.

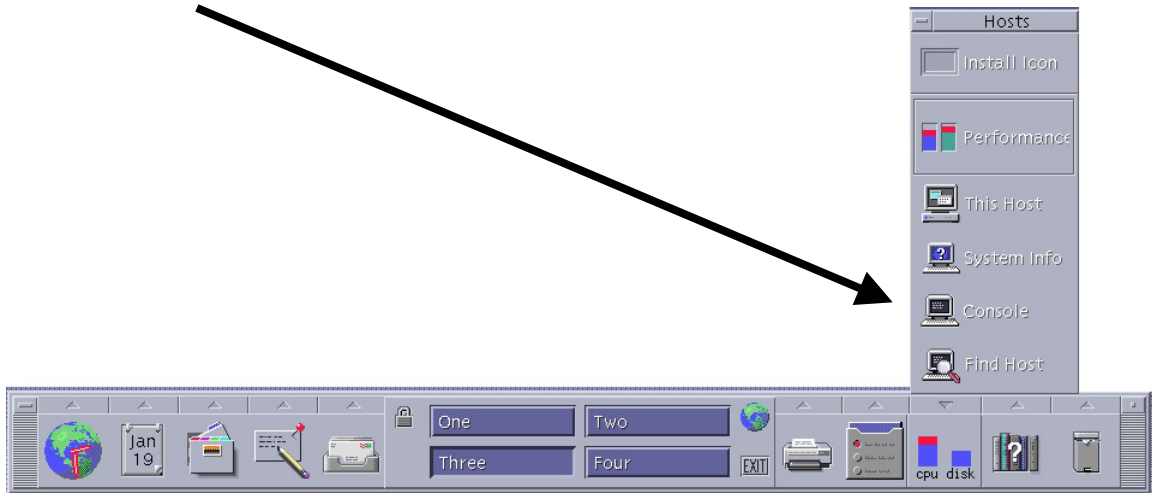
One of the sources of greatest confusion for Unix novices is that all Unix systems can act as hosts. That is, they can become file servers. Someone can be using a Unix system as a workstation and others can be using that system via remote login, Telnet, FTP or X-Terminal sessions from some other device on the network.

Starting a console session from the Solaris OS GUI

1) click on the Hosts menu icon (third up-arrow from the bottom right of the screen)



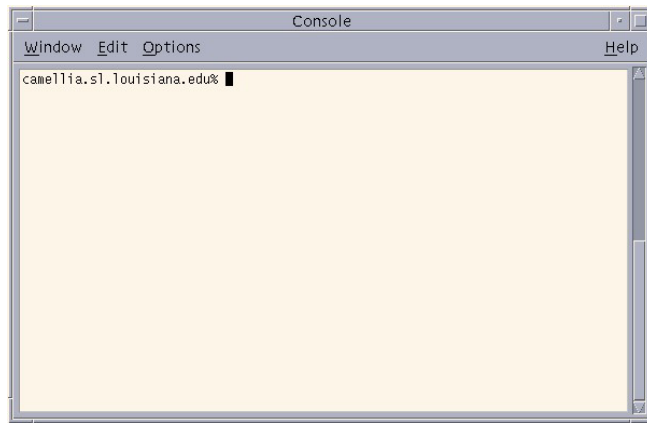
2) click on Console option



3) the Console window will appear

4) commands can be executed in the console window by typing in the command and pressing the Return or Enter key

5) note that commands in Unix are case sensitive



6) You can log out of Solaris by selecting the Exit button.



Common Unix Commands for the Command Line

1) clear
clears the console window

2) ls
list files in current directory (a directory is a folder for you Windows and Mac types)

```
d48.ucs.louisiana.edu% ls
1507      cpp2.cpp    cs1507      Mail        onei        toymom.cc
300      cs1501     dead.letter News        remote     toymom.cpp
cpp1.cpp  cs1502     DeadLetters nsmail      seveni     twoi
d48.ucs.louisiana.edu% █
```

3) `ls -l`

- a) written: `ls<space><minus>l`
- b) lists files in current directory in long form (gives more information)

4) `ls -a`

- a) written: `ls<space><minus>a`
- b) lists all files in current directory, including files normally hidden from view

5) `pwd`

displays the path to the current (working) directory

```
d35.ucs.louisiana.edu% pwd
/home/fdd5501
d35.ucs.louisiana.edu% █
```

6) `cd path-to-directory`

change directory – changes the current (working) directory to another directory given in the *path-to-directory*

```
d35.ucs.louisiana.edu%
d35.ucs.louisiana.edu% cd cs1501
d35.ucs.louisiana.edu% pwd
/w1/cs1501/fdd5501
d35.ucs.louisiana.edu%
```

7) `cd`

returns to your home directory

```
d35.ucs.louisiana.edu% cd
d35.ucs.louisiana.edu% pwd
/home/fdd5501
d35.ucs.louisiana.edu% █
```

8) `cd ..`

change directories to the parent directory of the current (working) directory

```
d35.ucs.louisiana.edu%
d35.ucs.louisiana.edu% pwd
/home/fdd5501/exampleDir
d35.ucs.louisiana.edu% cd ..
d35.ucs.louisiana.edu% pwd
/home/fdd5501
d35.ucs.louisiana.edu% █
```

9) `rm path-to-file`

- a) remove (delete) the file given by the path
- b) example: `rm myProg.cc`

10) `cp path-to-file target-path`

- a) copy the file given by *path-to-file* to the target directory or file name given by *target-path*
- b) example of copying a file from a home directory to a CMPS 150 project directory for section 8: `cp hwk54.cc cs1508`
- c) example of copying a file into the current directory: `mv cs150/myfile.cc .`

11) `mv path-to-file target-path`

- a) move the file given by *path-to-file* to the target directory or file name given by *target-path*; often used to rename a file
- b) example of using mv to rename a file: `mv myProg.cpp hmwk76.cc`

12) `cat path-to-file`

- a) written: `cat<space><the path to a file>`
- b) displays the contents of a text file on the screen

```
d35.ucs.louisiana.edu% cat cpp1.cpp
#include <iostream>
#include <cmath>

using namespace std;

// declaration of constants for program input
const float a = 12;
const float b = 233;
const float c = 2;

void main ( )
{
    // declaration of variables
    float pos, neg;

    // processing of equations
    pos = (-b + sqrt(b*b - 4.0*a*c))/(2.0*a);
    neg = (-b - sqrt(b*b - 4.0*a*c))/(2.0*a);

    // output of results
    cout << "The two answers of produced from " << a << ", " << b << " and "
    << c
        << " are " << pos << " and " << neg << endl;
}

d35.ucs.louisiana.edu% █
```

13) `more path-to-file`

- a) like cat, but displays one screen at a time

14) `| more`

- a) written: `<vertical-dash>more`; the vertical dash is referred to as the “pipe” operator
- b) added to a command to display the output of the command one screen at a time
- c) for example: `ls -l | more`

15) `!<letter>`

- a) written: `!<a-character>`
- b) re-runs the last command that starts with the character given

```
d35.ucs.louisiana.edu% pwd
/home/fdd5501
d35.ucs.louisiana.edu% !p
pwd
/home/fdd5501
d35.ucs.louisiana.edu% █
```

Running Unix Programs Frequently Used in CMPS 150

- 1) `nedit filename &`
 - a) runs the editor Nedit in the background (this allows you to switch back and forth from the editor to the Console or X-Terminal window by mouse click)
 - b) can be run from the Console window or an X-Terminal session, but cannot be run from a simple Telnet session (in Telnet, use vi)

- 2) `emacs filename &`
 - a) runs the editor Emacs in the background (this allows you to switch back and forth from the editor to the Console or X-Terminal window by mouse click)
 - b) can be run from the Console window or an X-Terminal session, but cannot be run from a simple Telnet session (in Telnet, use vi)

- 3) `dtpad filename &`
 - a) runs the editor Desktop-Pad in the background (this allows you to switch back and forth from the editor to the Console or X-Terminal window by mouse click)
 - b) can be run from the Console window or an X-Terminal session, but cannot be run from a simple Telnet session (in Telnet, use vi)

- 4) `g++ file-of-code.cc`
 - a) `g++` is the GNU C++ compiler
 - b) compiles a file of C++ code and writes the resulting binary code into a software file called *a.out*; the resulting program may be run by typing *a.out*

- 5) `g++ -o file-to-place-binary-code-in file-of-code.cc`
 - a) `g++` is the GNU C++ compiler
 - b) compiles a file of C++ code named in *file-of-code.cc* and writes the resulting binary code into the software file named in *file-to-place-binary-code-in*; the resulting program may be run by typing the file named in *file-to-place-binary-code-in*

- 6) `submit -d file-to-submit`
 - c) submit the file named in *file-to-submit* for grading
 - d) when `submit` is run, you will be asked for the clid of your TA or Grader
 - e) when `submit` is run, you will be asked which assignment you are submitting; reply by typing "assn", then the number of the assignment; for example: `assn97`

- 7) `ctrl-z` and `bg`
 - a) `Ctrl-z` and `bg` are two commands
 - b) allow most programs to be run in background mode even after they are started
 - c) for example:
 - i) start `dtpad`
 - ii) click in the Console window
 - iii) type `Ctrl-z`
 - iv) type `bg`the program is now running in the background, allowing you to switch between the console and the program with a mouse click

Some UNIX Tutorial Resources Online

Introductory UNIX Tutorials

<http://www.isu.edu/departments/comcom/unix/workshop/unixindex.html>

<http://www.math.utah.edu/lab/unix/unix-tutorial.html>

<http://bromide.ocean.washington.edu/unix.tutorial.html>

<http://web.cs.mun.ca/unixtut/demo.html>

Introductory vi Tutorials

http://math.la.asu.edu/vi_tutorial/vi3.html

<http://www.eng.hawaii.edu/Tutor/vi.html>

<http://www.epcc.ed.ac.uk/tracs/vi.html>

<http://www.networkcomputing.com/unixworld/tutorial/009/009.html>