What is Unix?

- An operating system which runs on many different kinds of computers – Suns, HPs, DECs, PCs, Macs. Linux is a version of Unix.
- Users access the operating system via what is called a shell. You are probably using tcsh or bash. They are many available sh, csh, ksh, etc, though not all may be available on our system.
- Unix prompt, or shell prompt, is where you type your commands.
- The way you prompt looks depends on which shell you are running and how you configure it.
- Anything running on the machine (even the operating system itself) is called a process.
- Unix has been called confusing and a pain to learn, but it's also flexible and powerful.

Prompts

Look depends on which shell you are running and how you have things configured. This configuration can involve including this like which machine you are on, which directory you are in, time, date, process number, etc.

`<64> stat /home/irwin%` (system default)

`bash-2.05a$` (bash default)

`stat:~>` (mine. `set prompt="%m:%~> "`)

All commands are given to the shell are typed at the prompt.

This command might be a script you've written. You can write your own programs using Unix commands.

Basic Unix Commands

`ls` lists files in a directory

`ls` (lists file in current directory)

`ls -l unixcourse/*.s` (lists all files in directory unixcourse ending in .s in the long format)

```
fisher:~/statcomp> ls
EAgrant/           formexamples/       regentsgrant        unixcourse/
SurvCourse/        import.info         sitelicence.mac     webcontent
WWW/               important           sitelicence.pc      webpages
brian.phone        laptop.shili        sitelicence.sun     website

fisher:~/statcomp> ls -l unixcourse/*.s
-rw-r--r-- 1 irwin parstaff 20 Sep 27 14:53 unixcourse/temp.s
-rw-r--r-- 1 irwin parstaff 20 Sep 27 14:53 unixcourse/temp2.s
```

Files and directories whose names start with a "." aren't shown unless explicitly asked for with the –a option.

```
fisher:~/statcomp/unixcourse> ls -a
./             combined.s     temp2.s        tempdir2/
../            combined.txt   temp23.s       tempdir3/
.sdotfile      temp.s         tempdir/       weizens.pdf

fisher:~/statcomp/unixcourse> ls
combined.s     temp.s         temp23.s       tempdir2/     weizens.pdf
combined.txt   temp2.s        tempdir/       tempdir3/
```

`cd` change directory

`cd unixcourse` (goes to directory unixcourse located in the current directory)

`cd ~irwin/pub` (goes to my public directory)

`cd ..` (goes to parent directory)

Special Characters in file names:

~	home directory
~user	user's home directory
.	current directory
..	parent directory
/	directory break character
*	wildcard character – represents any character string
?	matches any single character

```
unix:~/statcomp/unixcourse> dir
total 120
drwxr-xr-x   5 irwin    parstaff    4096 Jun 26  2001 ./
drwxr-xr-x   7 irwin    parstaff    4096 Feb 18  2003 ../
-rw-r--r--   1 irwin    parstaff      20 Sep 28  2000 .sdotfile
-rw-r--r--   1 irwin    parstaff     204 Jun 26  2001 combined.s
-rw-r--r--   1 irwin    parstaff     204 Sep 27  2000 combined.txt
-rw-------   1 irwin    parstaff      20 Sep 27  2000 temp.s
-rw-r--r--   1 irwin    parstaff     184 Sep 27  2000 temp2.s
-rw-r--r--   1 irwin    parstaff      20 Sep 27  2000 temp23.s
drwxr-xr-x   2 irwin    parstaff    4096 Jun 26  2001 tempdir/
drwxr-xr-x   2 irwin    parstaff    4096 Jun 26  2001 tempdir2/
drwxr-xr-x   2 irwin    parstaff    4096 Sep 27  2000 tempdir3/

unix:~/statcomp/unixcourse> dir *.s
-rw-r--r--   1 irwin    parstaff     204 Jun 26  2001 combined.s
-rw-------   1 irwin    parstaff      20 Sep 27  2000 temp.s
-rw-r--r--   1 irwin    parstaff     184 Sep 27  2000 temp2.s
-rw-r--r--   1 irwin    parstaff      20 Sep 27  2000 temp23.s

unix:~/statcomp/unixcourse> dir temp*.s
-rw-------   1 irwin    parstaff      20 Sep 27  2000 temp.s
-rw-r--r--   1 irwin    parstaff     184 Sep 27  2000 temp2.s
-rw-r--r--   1 irwin    parstaff      20 Sep 27  2000 temp23.s

unix:~/statcomp/unixcourse> dir temp?.s
-rw-r--r--   1 irwin    parstaff     184 Sep 27  2000 temp2.s
-rw-r--r--   1 irwin    parstaff     184 Sep 27  2000 temp3.s
```

`rm` remove files

`rm -i temp3.s` (removes file temp3.s. The –i flag asks whether you really want to do this and is the default setup on our system. For most people typing rm temp3.s has the same result.)

`rm -r tempdir` (removes the directory tempdir and all file and directories contained in it)

```
fisher:~/statcomp/unixcourse> rm -i temp3.s
rm: remove temp3.s (yes/no)? y
```

`mv` move (rename) files and directories

`mv temp2.s newname.txt` (renames the file temp2.s to newname.txt)

`mv tempdir/* tempdir2` (moves all files in directory tempdir to directory tempdir2)

```
fisher:~/statcomp/unixcourse> ls tempdir
garbage

fisher:~/statcomp/unixcourse> ls tempdir2

fisher:~/statcomp/unixcourse> mv tempdir/* tempdir2

fisher:~/statcomp/unixcourse> ls tempdir

fisher:~/statcomp/unixcourse> ls tempdir2
garbage
```

`cp` copy files and directories

`cp temp2.s newname.txt` (copies the file temp2.s to newname.txt)

`cp tempdir/* tempdir2` (copies all files in directory tempdir to directory tempdir2)

```
fisher:~/statcomp/unixcourse> ls tempdir
garbage

fisher:~/statcomp/unixcourse> ls tempdir2

fisher:~/statcomp/unixcourse> cp tempdir/* tempdir2

fisher:~/statcomp/unixcourse> ls tempdir
garbage

fisher:~/statcomp/unixcourse> ls tempdir2
garbage
```

`mkdir` make a new directory off of the current directory

`mkdir newdir` (makes directory newdir)

`rmdir` removes a directory

`rmdir olddir` (removes directory olddir)

man print manual page

man man (prints manual page for command man)

man(1) man(1)

NAME
man - format and display the on-line manual pages
manpath - determine user's search path for man pages

SYNOPSIS
man [-acdfFhkKtwW] [--path] [-m system] [-p string] [-C config_file] [-M pathlist] [-P pager] [-S section_list] [section] name ...

DESCRIPTION
man formats and displays the on-line manual pages. If you specify section, man only looks in that section of the manual. name is normally the name of the manual page, which is typically the name of a command, function, or file. However, if name contains a slash (/) then man interprets it as a file specification, so that you can do man ./foo.5 or even man /cd/foo/bar.1.gz.

See below for a description of where man looks for the manual page files.

OPTIONS
-C config_file
Specify the configuration file to use; the default is /usr//etc/man.config. (See man.conf(5).)

-M path
Specify the list of directories to search for man pages. Separate the directories with colons. An empty list is the same as not specifying -M at all. See SEARCH PATH FOR MANUAL PAGES.

-P pager
Specify which pager to use. This option overrides the MANPAGER environment variable, which in turn overrides the PAGER variable. By default, man uses /usr/bin/less -isr.

-S section_list
List is a colon separated list of manual sections to search. This option overrides the MANSECT environment variable.

Note many commands take options, indicated by the dash. Checking a command's man page is a good way to find out about them.

Use –k option to search for commands. This can also be done with the `apropos` command.

```
stat:~> man -k manual
man                 (1)  - format and display the on-line manual pages
man [manpath]       (1)  - format and display the on-line manual pages
man2html            (1)  - format a manual page in html
perlxs              (1)  - XS language reference manual
wget                (1)  - GNU Wget Manual
whereis             (1)  - locate the binary, source, and manual page
files for a command

stat:~> apropos manual
man                 (1)  - format and display the on-line manual pages
man [manpath]       (1)  - format and display the on-line manual pages
man2html            (1)  - format a manual page in html
perlxs              (1)  - XS language reference manual
wget                (1)  - GNU Wget Manual
whereis             (1)  - locate the binary, source, and manual page
files for a command
```

`cat` prints a file to the screen or concatenatesfiles together

`cat temp.s` (prints to contents of the file temp.s to the screen)

```
fisher:~/statcomp/unixcourse> cat temp.s
this is a test file
```

`cat temp.s temp2.s` (prints the files temp.s and temp2.s one after the other to the screen)

```
fisher:~/statcomp/unixcourse> cat temp.s temp2.s
this is a test file
this is another test file for the demo
The quick brown fox jumps over the lazy dog.
Cozy lummox gives smart squid who asks for job pen.
What do these two sentences have in common?
```

`cat temp.s temp2.s > combined.s`

(concatenates the files temp.s and temp2.s and writes the result to a new file combined.s. The output does not appear on the screen.)

```
fisher:~/statcomp/unixcourse> cat temp.s temp2.s > combined.txt
fisher:~/statcomp/unixcourse>
```

`more` pages through a file on the screen

`less` an enhanced version of more where you can also go backward. Not available on all systems.

Input, Output and Redirection

- Input usually comes from Standard Input (keyboard)
- Output usually goes to Standard Output (screen)
- Can change one or both of these

Basic command format

```
command -options <inputfile >outputfile
```

`tr s S < temp2.s` (replaces every s with a S in the file temp2.s and prints the output to the screen)

`tr s S < temp2.s > temp2.trans` (replaces every s with a S in the file temp2.s and puts the output in the file temp2.trans)

`cat temp.s temp2.s > combined.s` (concatenates the files temp.s and temp2.s and writes the result to a new file combined.s.)

>> instead of > will append output to the file.

& at the end of a command line will run the job in the background. This allows for jobs to be run while you are doing other things.

Pipes |

A method for stringing commands together

> `frm | tail` (prints where the last 10 e-mail in my inbox are from)
>
> `frm` (prints where all e-mails in my inbox are from)
>
> `tail` (prints the last 10 lines of a file)

The pipe takes the output on the standard output and transfers it to the standard input for the next command.

Much nicer than doing something like

```
frm > tempfile

tail < tempfile

rm tempfile
```

The output from the last command can be redirected

```
frm | grep Brian | grep Smith > BSmithFrm
```

`alias` allow for creation of new commands or for commands to redefined. The form of the alias command depends on your shell. Aliases are defined differently for tcsh/csh and bash/sh.

`alias copy cp` (tcsh)

`alias copy=cp` (bash)

(copy acts the same as the command cp)

`alias frmtail 'frm | tail'` (tcsh)

`alias frmtail 'frm | tail'` (bash)

(command frmtail prints where the last 10 e-mails are from)

`alias dir 'ls -alF \!* | more'` (tcsh)

(command dir lists all of the given files and directories in the long format one screen at a time)

I'm not sure how to do this with bash, but apparently it involves shell functions

`alias rm` (shows current alias for rm command)

```
fisher:~> alias rm
rm -i
```

`alias` (shows all defined aliases)

`unalias` removes alias from system or sets command back to its default settings.

`unalias rm` (sets rm back to its default)

`unalias dir` (removes dir as a command)

Customizing your setup (.login, .cshrc (.or .bashrc),& .logout)

- Don't want to have to retype desired aliases and so on every time you log in.
- Can place the commands in your .login and .cshrc (or .bashrc) files
- .login should contain the commands you want run each time you login. For example, date and time info, new email listings, printing out who is on.
- .cshrc should contain commands you want available in every shell. For example, setting aliases and environment variables. Anything that generates output should go in .login.
- Place commands in the .logout file for anything you wish done before logging out.

File Permissions

It is possible to control who can see your files. For example everybody in the department can examine most of my files. I believe that for most of you, only you can see your files.

`umask` sets the default permissions

`umask 022` (anybody can read your files, only you can edit them)

`umask 077` (only you can read your files, plus edit them – system default)

If you wish to change this, put it in your .cshrc file

`chmod` set the permission for any file

`chmod ugo±rwx filename`

u = user, g = group, o = others

+ = add permission, – = remove permission

r = read, w = write, x = execute

`chmod go-rw temp` (removes read and write permissions for group members and others for file temp)

`chmod go+r temp` (gives read permission back to group members and others for file temp)

There is a second form of the chmod command which allows the permission to be set instead of adding or remove permission at a certain level

`chmod ubo filename`

u = user , g = group, o = others

+ = add permission, – = remove permission

r = 4 if read allowed, 0 otherwise

w =2 if write allowed, 0 otherwise

x = 1 if execute allowed, 0 otherwise

u, g, and o have to be set to r+w+x fore each

`chmod 644 filename` (anybody can read your files, only you can edit them)

`chmod 755 directory` (anybody can access directory, only you can alter it)

`chmod 600 filename` (only you can read your files, plus edit them – system default(?))

`chmod 700 directory` (only you can access the directory – system default(?))

Current permissions can be determined with the ls –l command

```
fisher:~/statcomp/unixcourse> ls -l
total 18
-rw-r--r-- 1 irwin parstaff 499 Sep 28 11:56 Brian.Smith.mail
-rw-r--r-- 1 irwin parstaff 204 Sep 27 16:22 combined.txt
-rw-r--r-- 1 irwin parstaff 20 Sep 27 14:53 temp.s
-rw-r--r-- 1 irwin parstaff 184 Sep 27 17:33 temp2.s
-rw-r--r-- 1 irwin parstaff 20 Sep 27 15:47 temp23.s
-rw-r--r-- 1 irwin parstaff 20 Sep 27 15:06 temp3.s
drwxr-xr-x 2 irwin parstaff 512 Sep 27 15:24 tempdir/
drwxr-xr-x 2 irwin parstaff 512 Sep 28 10:58 tempdir2/
drwxr-xr-x 2 irwin parstaff 512 Sep 27 15:23 tempdir3/
```

First character indicate type of entry

- = file, d = directory, l = symbolic link

Characters 2 – 4 give user status

Characters 5 – 8 give group status

Characters 9 – 11 give other status

r = read, w = write, x = execute,

- = permission not given

Note for a directory to be readable, it also needs execute status.

Printing

`lp` sends a text file or a postscript file to the printer

`enscript` prints text files with nice layouts. Allows files to be printed sideways or in two columns

`enscript -dlj temp2.s` (prints file temp2.s on printer in 711. Both commands take this –d option which allows an alternate printer to be selected. `lj3` I believe is the printer in 606e)

Logging in and out

`logout` logs you out of the system

`su` switches user

`login` logs you in as a different user

`ssh` allows you to log onto to a different machine (such as fas or ice)

Text Editors

`vi` oldest

`pico` easiest to use

`emacs` most powerful. Can customize it and run other programs from it. Useful for running Splus and R.

Document Processing

`tex`	A high end document processing system, particularly useful with mathematics. Uses a text markup language.
`latex`	tex with extensions to make it friendlier to use.
`xdvi`	display dvi file (only under Xwindows). A dvi file is the result of running `tex` or `latex`
`dvips`	prints dvi file or creates postscript file
`dvipdf`	creates pdf file from dvi file
`latexpdf`	creates pdf file directly from latex file

File Convertors

`ps2pdf`	converts postscript to pdf (generally ps2*)
`pdf2ps`	converts pdf to postscript (generally pdf2*)

Graphics (only under Xwindows)

`gv`	postscript previewer
`acroread`	Adobe Acrobat reader
`gimp`	GNU Image Manipulation Program

E-mail

`pine` recommended e-mail program. Easy to use with default setup, but easily customizable. Uses pico as the text editor.

`mail` ancient program- unfriendly

Statistics and Mathematics Packages

`Splus` most popular statistics package in academics (stat: version 5.1, ice: versions 3.4 [default], 5.1 [`Splus5`])

`mathematica` symbolic math, graphics, arbitrary precision calculations (ice. `math` is text version)

`matlab` matrix math (ice: versions 5.3 [default], 6.1 [`/usr/local/matlab-6.1/bin/matlab`]

`R` an Splus clone (stat: version 1.6.2)

Programming

`cc, gcc` C compliers

`g++` C++ compiler

`f77, g77` Fortran 77 compiler

`perl` Practical Extraction and Report Language

Information sources

- Fiamingo FG, DeBula L, and Condron L (1998). Introduction to Unix. <http://wks.uts.ohio-state.edu/unix_course/unix_book.pdf>. Will be available from class web page. This document also has a fairly lengthy list of other unix references.

- Taylor, Christopher C. Unix is a Four Letter Word <http://unix.t-a-y-l-o-r.com/4ltrwrd.pdf>

- Lasser, Jon. *Think UNIX*. QUE Corporation

- Levine, John R. and Margaret Levine Young. *UNIX for Dummies*. Dummies Press, a division of IDG Books Worldwide, Inc.

- Martin, Don, et al. *UNIX Primer Plus*. Waite Group Press, A Division of Macmillan USA, Inc.

- Ray, Deborah S. and Eric J. Ray. *Visual Quickstart Guide: UNIX*. Peachpit Press.

- Reichard, Kevin and Eric Foster-Johnson. *UNIX in Plain English*. M&T Books.

- Lamb, Linda and Arnold Robbins. *Learning the vi Editor*. O'Reilly

- Pedersen, Jesper, et al. *SAMS Teach Yourself Emacs in 24 Hours*. Sams, A Division of Macmillan Computer Publishing.