Lesson Module Checklist

- Slides –
- Flash cards –
- Page numbers -
- 1st minute quiz –
- Web Calendar summary –
- Web book pages -
- Commands –
- Lab tested –

- CCC Confer wall paper / quiz emailed-
- enlightenment script tested –
- pick up Polycom phone

- Check that headset is charged –
- Wireless lapel mic backup battery -
- Backup slides, CCC info, handouts on flash drive -

CIS 90 - Lesson 4

Instructor: **Rich Simms**
Dial-in: **888-450-4821**
Passcode: **761867**


Dieskau

Jonathan

Ana

David


Obie

Dave

Cole

Corey

Nancy

Ryan

Elia


Tasha

Darren

Scott


Devin

Everett

Juan

Raven

Rogan


Mike


Mook


Melissa

Cameron

Jose

Jeff

Matt

Kenneth

Ousmane

Ian

TBD

TBD

TBD

TBD

TBD

Email me (risimms@cabrillo.edu) a relatively current photo of your face for 3 points extra credit

First Minute Quiz

Please answer these questions **in the order** shown:

email answers to: risimms@cabrillo.edu

[ ] Has the phone bridge been added?

[ ] Is recording on?

[ ] Does the phone bridge have the mike?

[ ] Share slides, putty (rsimms, simben90, rodduk90), Chrome, VLab

[ ] Disable spelling on PowerPoint

The UNIX/Linux File System

Objectives	Agenda
• Become familiar with the UNIX file hierarchy. • Be able to navigate the hierarchy using cd, ls and pwd commands. • Understand the key elements of a file. • Be able to distinguish the different UNIX files types. • Learn appropriate commands to view file contents.	• Quiz • Questions & Review • Unix files • Viewing files • The UNIX Directory Hierarchy • Navigating the file system • File types • Exercise: Enlightenment • Wrap up

Previous material and assignment

1. Questions on previous material?

2. Lab 3 questions?

   - I'll use check3 for grading
   - bash shell vs mail "shell"
   - clean up duplicates before last submittal
   - mail -f, mail -f mbox, mail -f uhistory

Lab 2
Post Mortem

Lab 2 Results

1) show shell	(0)
2) type commands	xxxxx (5)
3) echo variables	xx (2)
4) set TERM	xxxxxxxx (8)
5) upper/lower case	(0)
6) who –g	xxx (3)
7) number of arguments	xxxxxxxxxxxx (12)
8) CR and quotes	xxxxxxxx (8)
9) ; to separate commands	xxxxx (5)
10) change password	(0)
11) uname options	xxxxxxxxxxxx (12)
12) banner	xx (2)
13) finger	xxx (3)
14) id	x (1)
15) man	(0)
16) whatis vs man –f	xxx (3)
17) Tryme	xxxxxxxxxxx (11)
18) who –q	xxxxxxxxx (9)
19) man –k vs apropos	xxxxxxxxxxxxxxxxxxx (19)
20) info bash	x (1)
21) Google	(0)
22) sqrt	xxxxx (5)
Q1 – input from cmd line	xxxxxxx (7)
Q2 – input from keyboard	xxxxxxxxxx (10)
Q3 – input from OS	xxxx (4)

Each x = 1 point missed

Lab 2 Results – Q2

2. Use the following commands as arguments to the type command, to find out where each of the commands resides.

cmd argument
type man
type uname
type tryme
type echo
type type

Lab 2 Results – Q2

```
/home/cis90/simben $ type man
man is /usr/bin/man
```

*The **man** command is in the /usr/bin directory*

```
/home/cis90/simben $ type uname
uname is /bin/uname
```

*The **uname** command is in the /bin directory*

*Use the **type** command to find where on the path a command is located*

```
/home/cis90/simben $ type tryme
tryme is /home/cis90/simben/bin/tryme
```

*The **tryme** command is in the bin/ directory of our home directory*

```
/home/cis90/simben $ type echo
echo is a shell builtin

/home/cis90/simben $ type type
type is a shell builtin
```

*The **echo** and **type** commands are built into the bash shell*

Lab 2 Results – Q7

7. How many arguments do each of the following command lines have?

```
echo one      two                 threefour
echo "My TERM type is " $TERM
echo one.two.three
```

Lab 2 Results – Q7

```
/home/cis90/simben $ echo one       two              threefour
one two threefour
```
(3 arguments)

```
/home/cis90/simben $ echo "My TERM type is " $TERM
My TERM type is  xterm
```
(2 arguments)

```
/home/cis90/simben $ echo one.two.three
one.two.three
```
(1 argument)

Lab 2 Results – Q8

8. What is the difference in output between the following two commands? Note, the $ and > are part of the prompt, you don't need to type them.

```
$ echo red 'white
> and blue'
```

and

```
$ echo red white \
> and blue
```

Note: the [enter] key is pressed immediately after the last character of each line

Lab 2 Results – Q8

```
/home/cis90/simben $ echo red 'white<newline>
> and blue'
red white
and blue
```

The unclosed single quote prevents the <newline> from signaling the end of the command.

The <newline> gets passed to the echo command.

```
/home/cis90/simben $ echo red white \<newline>
> and blue
red white and blue
```

The <newline> is escaped in this example. The shell ignores it and continues to prompt the user for the rest of the command.

The escaped <newline> is NOT passed to the echo command.

Pressing the Enter (or Return on Macs) key generates an invisible <newline> metacharacter.

This signals the shell to stop prompting and process the command line.

Lab 2 Results – Q8

Note: Primary prompt is determined by the value of PS1

```
/home/cis90/simben $ echo $PS1
$PWD $
```

The value of the PWD environment variable is your current working directory

```
/home/cis90/simben $ echo red 'white
> and blue'
red white
and blue
```

Note: Secondary prompt is determined by the value of PS2

```
/home/cis90/simben $ echo $PS2
>
```

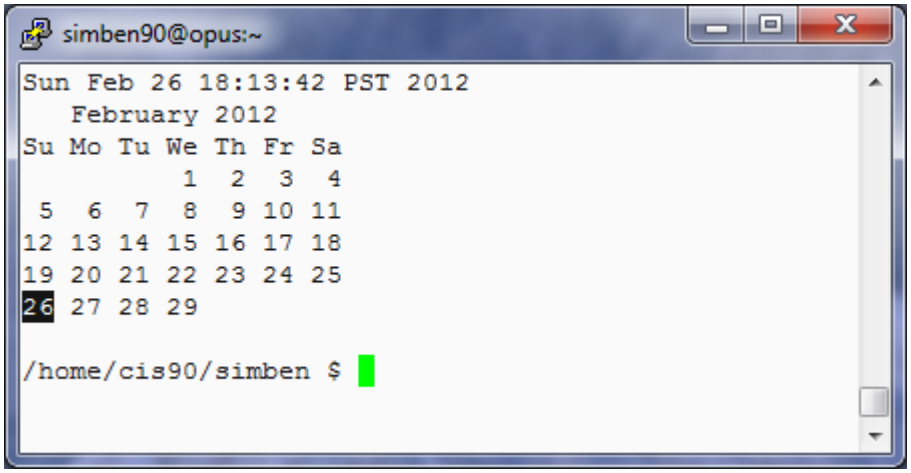
Lab 2 Results – Q9

9. Use the shell metacharacter ";" to write out a one line command that will clear the screen, print out the date and the current month's calendar.

$ _____

Lab 2 Results – Q9

```
/home/cis90/simben $ clear; date; cal
```



The ; metacharacter allows multiple commands on one line

Lab 2 Results – Q11

11. Using the **uname** command what options would you use to display just the operating system, it's kernel release numbers and the machine's network node hostname?

(Hint: Use the **man uname** command)

Lab 2 Results – Q11

Output from **man uname**

```
-a, --all
       print all information, in the  following  order,  ex
       omit -p and -i if unknown:

-s, --kernel-name
       print the kernel name

-n, --nodename
       print the network node hostname

-r, --kernel-release
       print the kernel release

-v, --kernel-version
       print the kernel version

-m, --machine
       print the machine hardware nam

-p, --processor
       print the processor type or "u

-i, --hardware-platform
       print the hardware platform or

-o, --operating-system
       print the operating system

--help display this help and exit
```

Use the man page to determine the options to show just the operating system, it's kernel release numbers and the machine's network node hostname

```
/home/cis90/simben $ man uname

/home/cis90/simben $ uname -orn
opus.cabrillo.edu 2.6.18-164.el5 GNU/Linux
```

or

```
/home/cis90/simben $ uname -o -r -n
opus.cabrillo.edu 2.6.18-164.el5 GNU/Linux
```

Use q to quit the man page

Free Software = Freedom to view and modify the source code

Richard Stallman started the GNU project in 1983 to create a free UNIX-like OS. He Founded the Free Software Foundation in 1985. In 1989 he wrote the first version of the GNU General Public License

```
/home/cis90/simben $ uname -orn
opus.cabrillo.edu 2.6.18-164.el5 GNU/Linux
```

node hostname *kernel release* *OS*

*Dan M. didn't like the order the **uname** command printed the information so he downloaded the source code, modified it, recompiled it. He now has his own version of the **uname** command!*

```
cis90@eko-04:~/dan/coreutils-7.4/src$ ./uname -orn
GNU/Linux 2.6.32-27-generic eko-04
```

OS *kernel release* *node hostname*

See forum post topic "Lab #2...even though 'info uname' output states". This is one of the really cool things about Linux and the GNU General Public License … if you don't like something about it you can change it!

http://opus.cabrillo.edu/forum/viewtopic.php?f=31&t=683&p=2632

Lab 2 Results – Q16

16. What is the **whatis** command? Use the command with the argument, bc

How does this compare to using the man command with -f option?

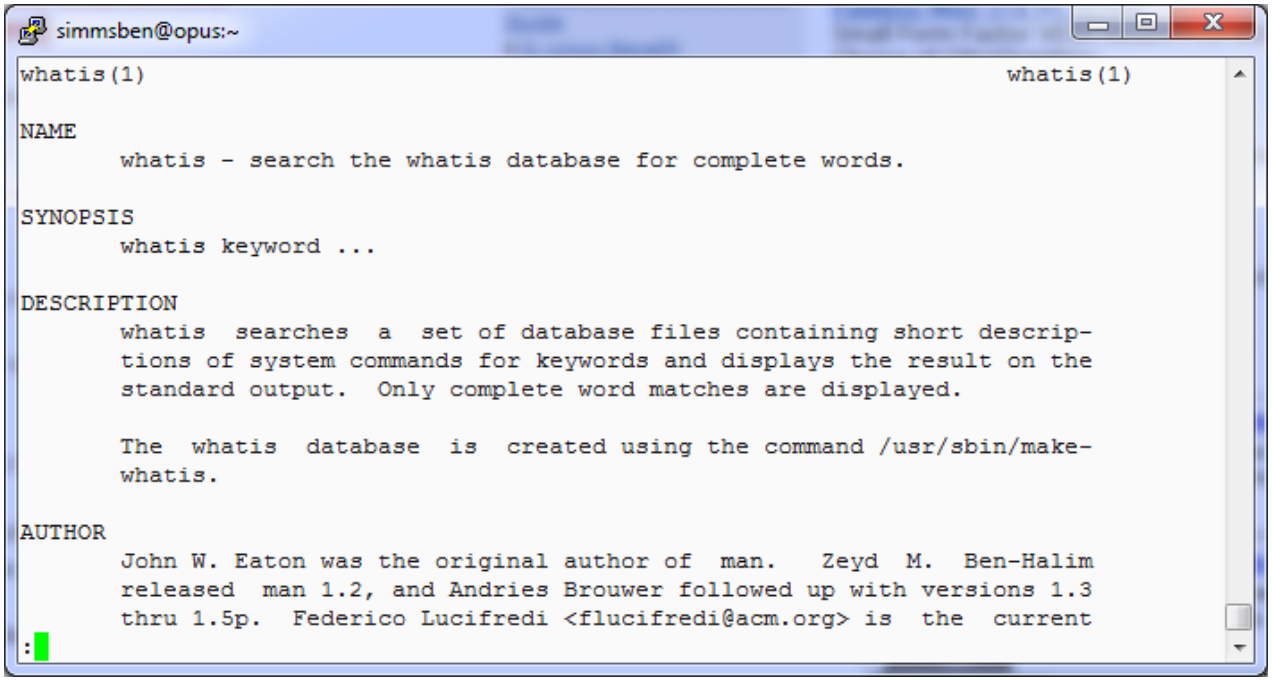
man -f bc

Lab 2 Results – Q16

*Use the **whatis** or **man** command to determine what the **whatis** command does.*

```
/home/cis90/simben $ whatis whatis
whatis                (1)  - search the whatis database for complete words

/home/cis90/simben $ man whatis
```

Output from **man whatis**



Lab 2 Results – Q16

*Use the **whatis** to find out about the BC command*

```
/home/cis90/simben $ whatis bc
bc                    (1)  - An arbitrary precision calculator language
bc                    (1p)  - arbitrary-precision arithmetic language
bc                   (rpm) - GNU's bc (a numeric processing language)
and dc (a calculator).
```

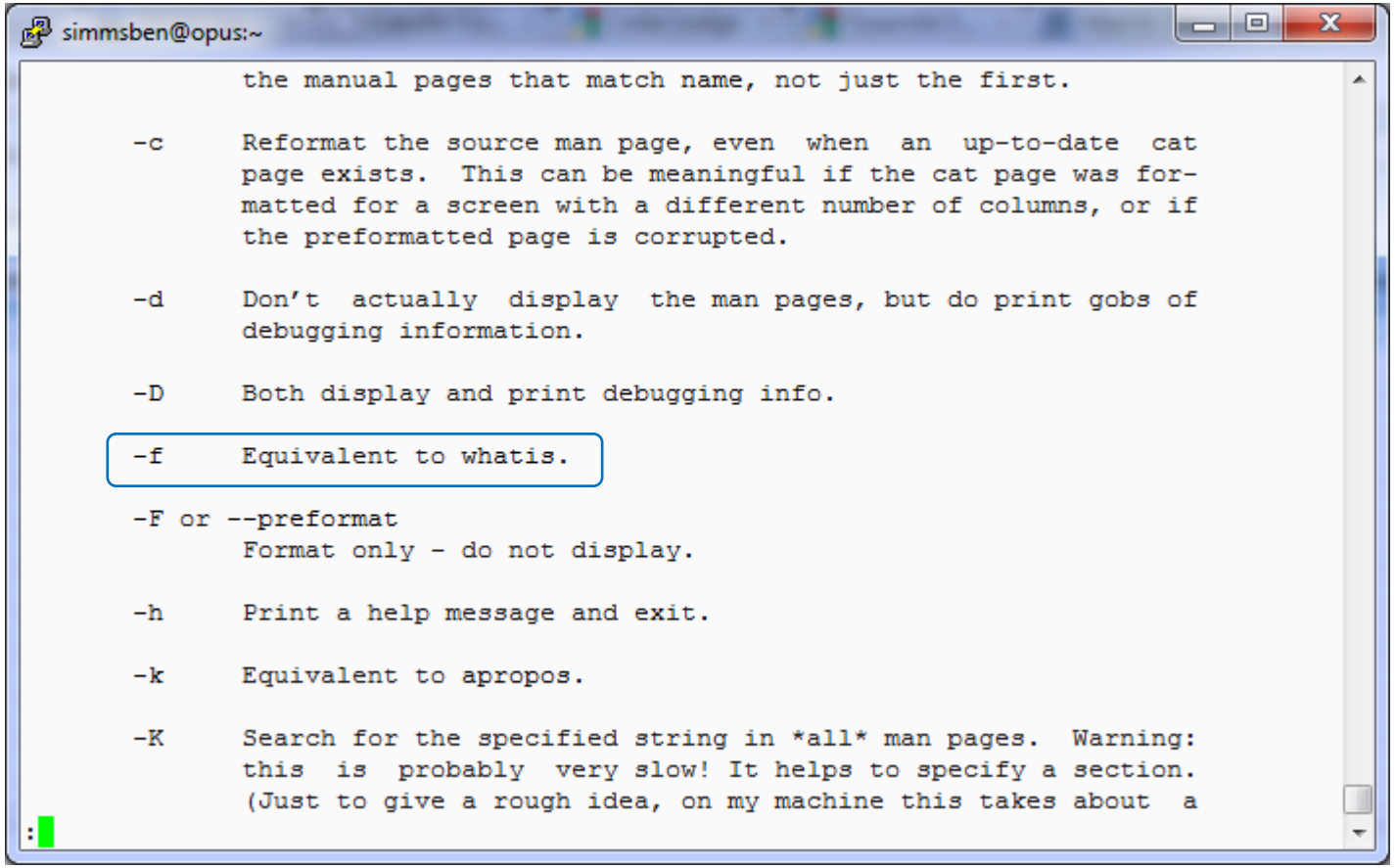
*Compare output with **man –f** command*

```
/home/cis90/simben $ man -f bc
bc                    (1)  - An arbitrary precision calculator language
bc                    (1p)  - arbitrary-precision arithmetic language
bc                   (rpm) - GNU's bc (a numeric processing language)
and dc (a calculator).
/home/cis90/simben $
```

They are equivalent

Lab 2 Results – Q16

Output from **man man**



***man man** will display the manual page for the man command and its documented there that the –f option is "Equivalent to whatis"*

Lab 2 Results – Q17

17. Is tryme a UNIX command? How do you know?

Lab 2 Results – Q17

```
/home/cis90/simben $ tryme
My name is "tryme"
I am pleased to make your acquaintance, Benji Simms
/tmp

/home/cis90/simben $ whatis tryme
tryme: nothing appropriate

/home/cis90/simben $ man tryme
No manual entry for tryme
```

UNIX commands are documented with man pages and have entries in the whatis database. ***tryme*** *does not appear in either one so is not a UNIX command*

Lab 2 Results – Q17

```
/home/cis90/simben $ type tryme
tryme is /home/cis90/simben/bin/tryme
```

__type__ shows __tryme__ resides in the bin/ directory of Benji's home directory

```
/home/cis90/simben $ file /home/cis90/simben/bin/tryme
/home/cis90/simben/bin/tryme: Bourne-Again shell script text executable
```

__file__ shows __tryme__ is a bash shell script

```
/home/cis90/simben $ cat  /home/cis90/simben/bin/tryme
#!/bin/bash
```

__cat__ shows the actual __tryme__ script itself

```
hello()
{
        cd /tmp
}
PATH=/bin
echo My name is \"`basename $0`\"
IFS=:
set `grep $LOGNAME /etc/passwd`
echo I am pleased to make your acquaintance, $5
hello
pwd
```

Lab 2 Results – Q18

18. Use the manual pages, and the **who** command, to find out the number of users logged on.

Lab 2 Results – Q18

Output from **man who**

```
       --lookup
              attempt to canonicalize hostnames via DNS

       -m     only hostname and user associated with stdin

       -p, --process
              print active processes spawned by init

       -q, --count
              all login names and number of users logged on

       -r, --runlevel
              print current runlevel

:
```

*The man page for **who** shows the q option will count the users logged in*

```
[rsimms@opus ~]$ who -q
helrog90 jimmel90 rsimms saljac193 vascar193
# users=5
```

Lab 2 Results – Q19

19. Run the command: **man -k boot** Use the manual pages to find out what the -k option does. What command is **man -k** equivalent to? Run the equivalent command and verify.

Lab 2 Results – Q19

Output from **man man**

simmsben@opus:~

```
       -d     Don't  actually  display  the man pages, but do print gobs of
              debugging information.

       -D     Both display and print debugging info.

       -f     Equivalent to whatis.

       -F or --preformat
              Format only - do not display.

       -h     Print a help message and exit.

       -k     Equivalent to apropos.

       -K     Search for the specified string in *all* man pages.  Warning:
              this  is  probably  very slow! It helps to specify a section.
              (Just to give a rough idea, on my machine this takes about  a
              minute per 500 man pages.)

       -m  system
              Specify  an alternate set of man pages to search based on the
              system name given.

       -p  string
:
```

*Use **man man** to read the manual page for the **man** command*

*the **apropos** command is equivalent to the **man –k** command*

Lab 2 Results – Q19

Output from **apropos boot**

```
simmsben@opus:~
/home/cis90ol/simmsben $ apropos boot
ExtUtils::Mkbootstrap (3pm)  - make a bootstrap file for use by DynaLoader
boot-scripts [boot]  (7)  - General description of boot sequence
bootparam            (7)  - Introduction to boot time parameters of the Linux kernel
firstboot           (rpm) - Initial s
firstboot-tui       (rpm) - A text in
grub                (rpm) - GRUB - th
initrd               (4)  - boot load
kexec                (8)  - directly
mbchk                (1)  - check the
mkbootdisk           (8)  - creates a
mkbootdisk          (rpm) - Creates a
perlboot             (1)  - Beginner(
pxeboot              (8)  - Network B
pxeos                (8)  - PXEBoot C
ty
reboot               (2)  - reboot or
reboot [halt]        (8)  - stop the
rhgb                (rpm) - Red Hat G
sys-unconfig         (8)  - shell scr
syslinux            (rpm) - Simple ke
system-config-netboot (8)  - Network
system-config-netboot (rpm) - network
system-config-netboot-cmd (rpm) - net
/home/cis90ol/simmsben $
```

Output from **man -k boot**

```
simmsben@opus:~
/home/cis90ol/simmsben $ man -k boot
ExtUtils::Mkbootstrap (3pm)  - make a bootstrap file for use by DynaLoader
boot-scripts [boot]  (7)  - General description of boot sequence
bootparam            (7)  - Introduction to boot time parameters of the Linux kernel
firstboot           (rpm) - Initial system configuration utility
firstboot-tui       (rpm) - A text interface for firstboot
grub                (rpm) - GRUB - the Grand Unified Boot Loader.
initrd               (4)  - boot loader initialized RAM disk
kexec                (8)  - directly boot into a new kernel
mbchk                (1)  - check the format of a Multiboot kernel
mkbootdisk           (8)  - creates a stand-alone boot floppy for the running system
mkbootdisk          (rpm) - Creates a boot floppy disk for booting a system.
perlboot             (1)  - Beginner(aqs Object-Oriented Tutorial
pxeboot              (8)  - Network Booting Operating Systems Configuration Utility
pxeos                (8)  - PXEBoot Operating System description Configuration Utili
ty
reboot               (2)  - reboot or enable/disable Ctrl-Alt-Del
reboot [halt]        (8)  - stop the system
rhgb                (rpm) - Red Hat Graphical Boot
sys-unconfig         (8)  - shell script to reconfigure the system upon next boot
syslinux            (rpm) - Simple kernel loader which boots from a FAT filesystem
system-config-netboot (8)  - Network Booting Configuration Utility
system-config-netboot (rpm) - network booting/install configuration utility (GUI)
system-config-netboot-cmd (rpm) - network booting/install configuration utility
/home/cis90ol/simmsben $
```

*the **apropos** command is equivalent to the **man –k** command*

Review

Command Syntax

Shell prints this to prompt user to enter a command

Shell parses this command line

***Options** modify the behavior of the command*

***Arguments** are what the command works upon*

***Redirection** is covered later in the course*

Examples

```
/home/cis90/simmsben $
/home/cis90/simmsben $ ls
/home/cis90/simmsben $ ls -l
/home/cis90/simmsben $ ls -lt
/home/cis90/simmsben $ ls -lt Poems/
/home/cis90/simmsben $ ls -lt Poems/ bin/
/home/cis90/simmsben $ ls -lt Poems/ bin/ > mylist
```

***Spaces (blanks)** are used to separate the command, options and arguments.*

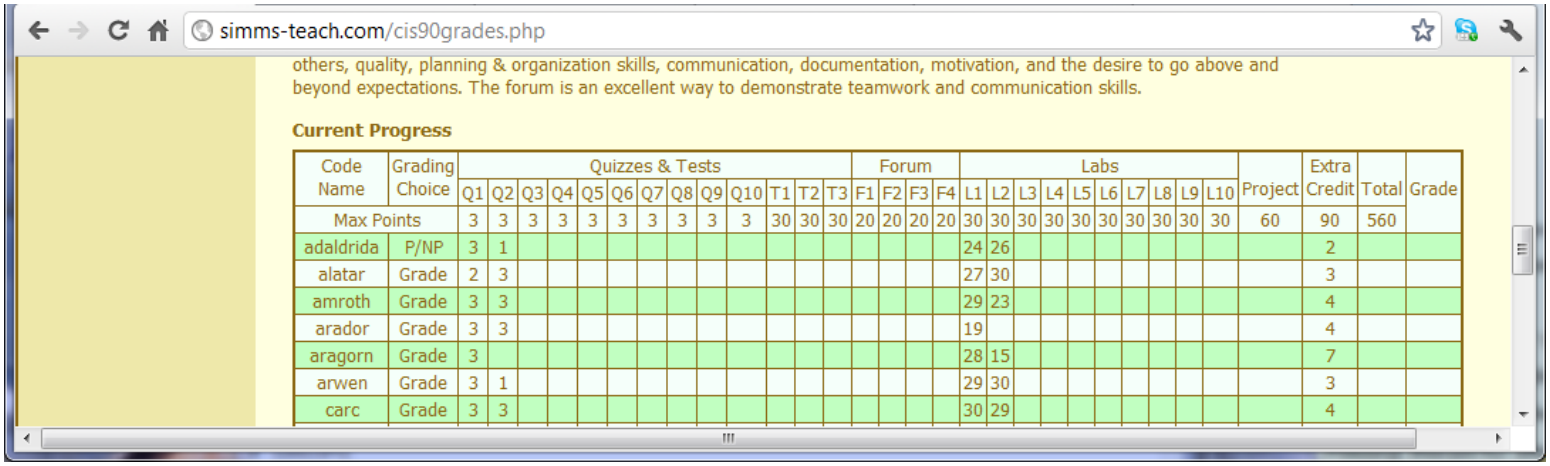
Housekeeping

- Grades posted on website

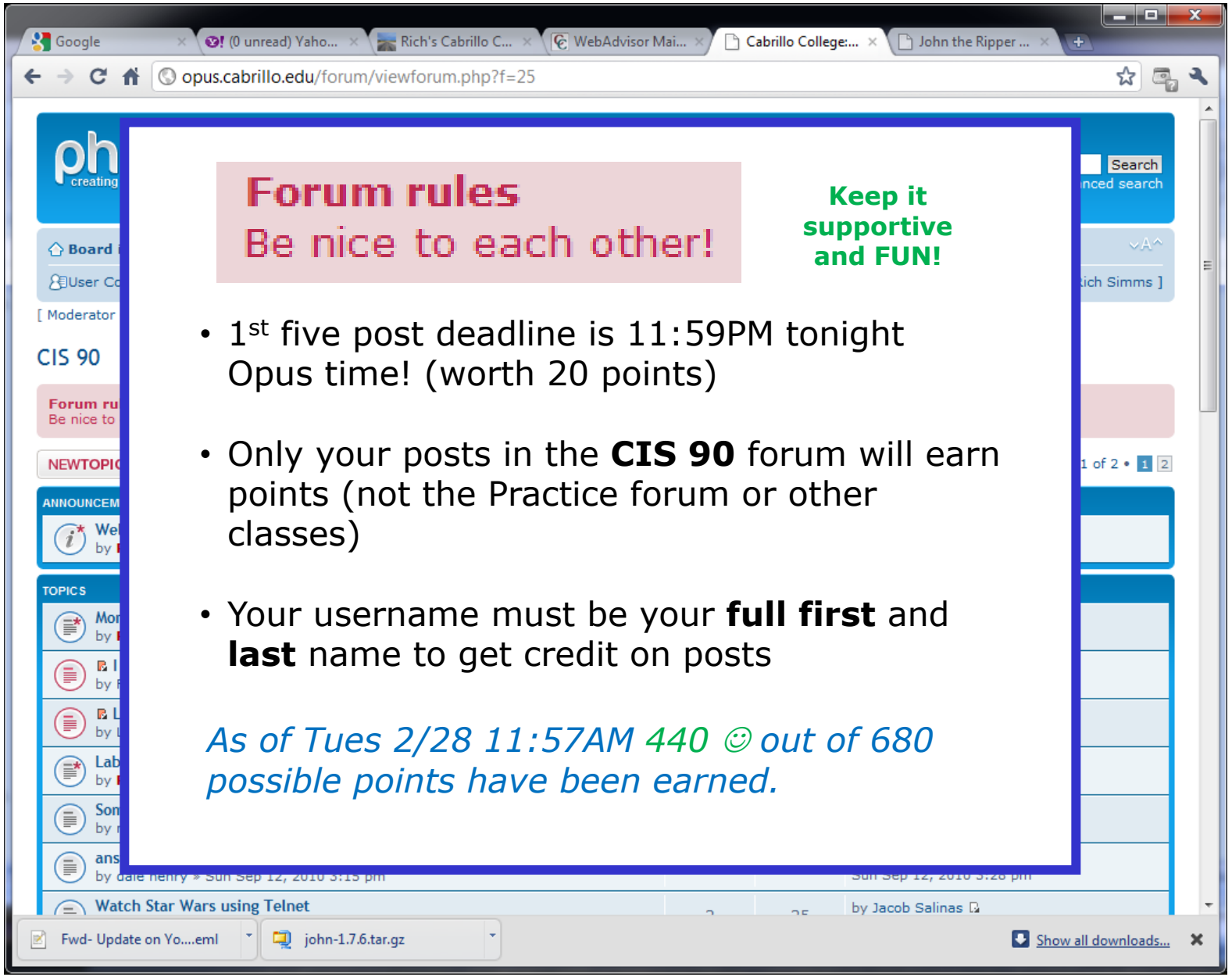
  (send me filled in student survey to get your grading code name,

  5 students have not turned in their surveys yet)

- Graded labs placed in your home directory
- Answers to labs in /home/cis90/answers/ directory
- Lab 3 and five forum posts due tonight at 11:59PM



Code Name	Grading Choice	Quizzes & Tests												Forum				Labs										Project	Extra Credit	Total	Grade
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	T1	T2	T3	F1	F2	F3	F4	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10				
Max Points		3	3	3	3	3	3	3	3	3	3	30	30	30	20	20	20	20	30	30	30	30	30	30	30	30	30	30	60	90	560	
adaldrida	P/NP	3	1																24	26										2		
alatar	Grade	2	3																27	30										3		
amroth	Grade	3	3																29	23										4		
arador	Grade	3	3																19											4		
aragorn	Grade	3																	28	15										7		
arwen	Grade	3	1																29	30										3		
carc	Grade	3	3																30	29										4		



Forum rules
Be nice to each other!

Keep it supportive and FUN!

- 1st five post deadline is 11:59PM tonight Opus time! (worth 20 points)
- Only your posts in the **CIS 90** forum will earn points (not the Practice forum or other classes)
- Your username must be your **full first** and **last** name to get credit on posts

As of Tues 2/28 11:57AM 440 ☺ out of 680 possible points have been earned.

UNIX Files

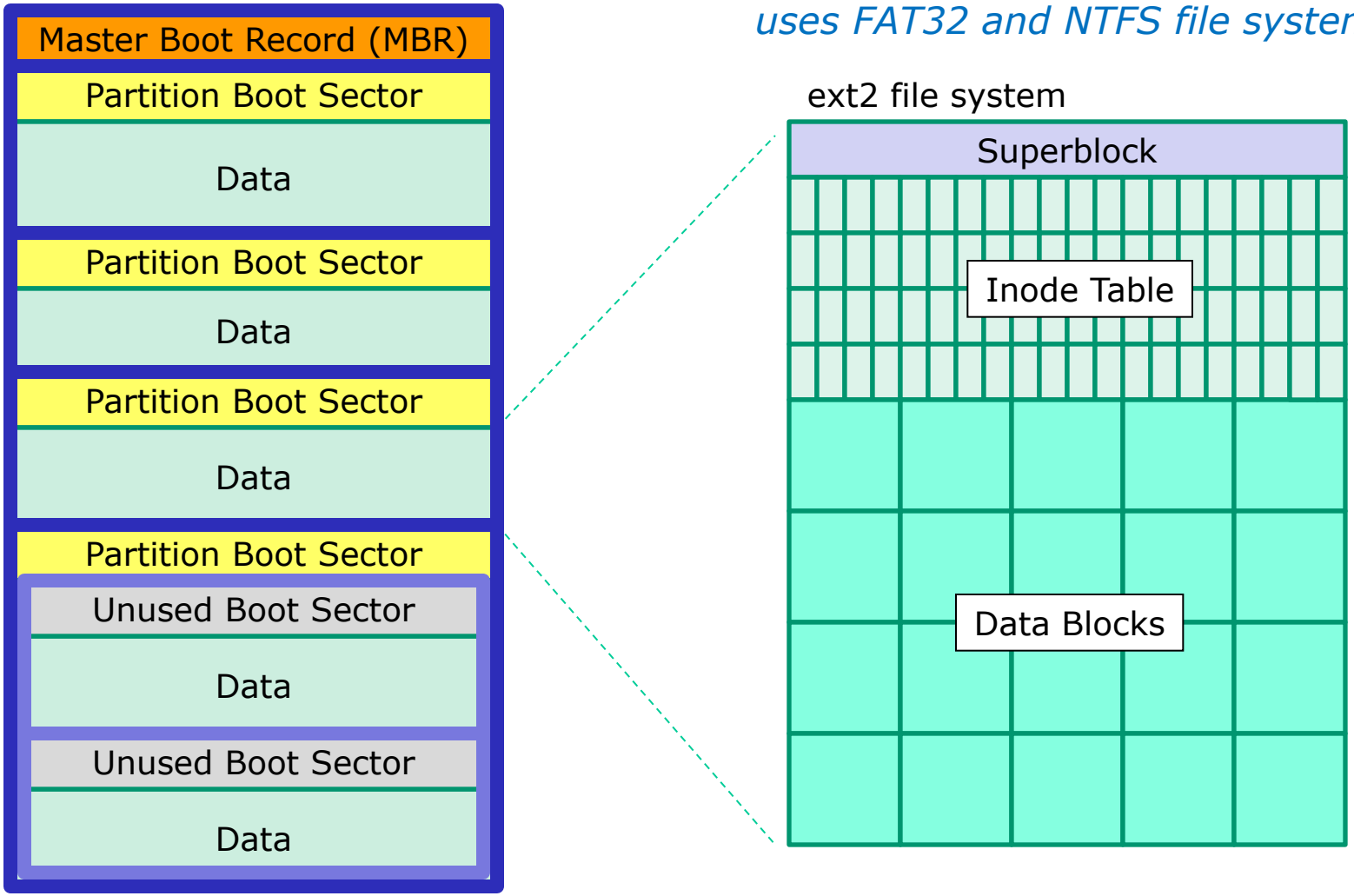
File Systems
Linux

A typical hard drive

This is where your files actually reside

File Systems
Linux

The hard drive is partitioned and the data areas can be formatted as a file system. Linux typically uses ext2, ext3 and ext4 file systems. Windows uses FAT32 and NTFS file systems.



UNIX Files
The three elements of a file

```
/home/cis90/simben/Poems $ ls
ant  Blake  nursery  Shakespeare  twister  Yeats

/home/cis90/simben/Poems $ ls -li twister
102625 -rw-r--r-- 1 simben90 cis90 151 Jul 20  2001 twister

/home/cis90/simben/Poems $ cat twister
A tutor who tooted the flute,
tried to tutor two tooters to toot.
Said the two to the tutor,
"is it harder to toot?  Or to
tutor two tooters to toot?"
```

name
+
inode
+
data

Note: filenames are stored in directories, **not** in inodes

bigfile 102559
bin 102560
letter 102594

Hello Mother! Hello Father!

Here I am at Camp Granada. Things are very entertaining,
and they say we'll have some fun when it stops raining.

All the counselors hate the waiters, and the lake has alligators. You remember Leonard Skinner? He got ptomaine poisoning last night after dinner.

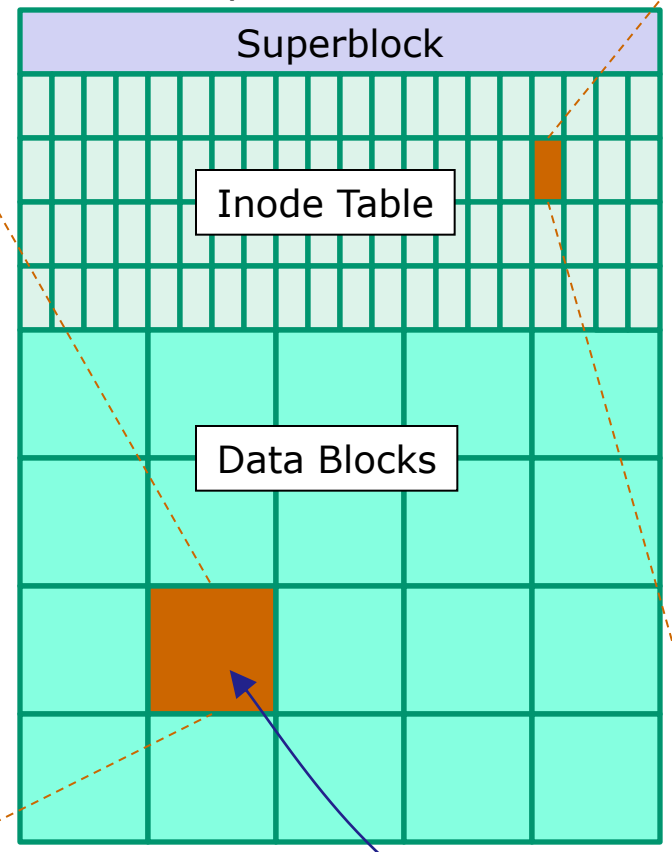
Now I don't want this to scare you, but my bunk mate has
malaria. You remember Jeffrey Hardy? Their about to organize a searching party.

Take me home, oh Mother, Father, take me home! I hate Granada.
Don't leave me out in the forest where I might get eaten
by a bear! Take me home, I promise that I won't make noise,
or mess the house with other boys, oh please don't make me
stay -- I've been here one whole day.

Dearest Father, darling Mother, how's my precious little brother? I will come home if you miss me. I will even let Aunt Bertha hug and kiss me!

Wait a minute! It's stopped hailing! Guys are swimming!
Guys are sailing! Playing baseball, gee that's better!
Mother, Father, kindly disregard this letter.

Alan Sherman



102594	inode number
-	Type
rw-r—r--	Permissions
1	Number of links
simben90	User
cis90	Group
1044	Size
2001-07-20	Modification time
2012-02-15	Access Time
2012-02-04	Change time
Pointer(s) to data blocks	Pointer(s) to data blocks

```
/home/cis90ol/simmsben $ ls -il letter
102594 -rw-r--r-- 1 simben90 cis90 1044 Jul 20  2001 letter
```

Unix Filename Conventions

UNIX File names conventions

Any combination of the following:

- Unix filenames are case sensitive

- Upper and lower case letters: **A-Z** and **a-z**
- Numbers: **0-9**
- Periods, underscores, hyphens: **. _ -**

- Don't use the following characters in filenames
  | ; , ! @ # $ ( ) < > / \ " ' ` ~ { } [ ] = + & ^
  <space> <tab>

Viewing Text Files

Important commands for your toolbox:

- cat *to print a file*
- more *to scroll down through a file*
- less *to scroll down and up a file*
- head *to print the beginning lines of a file*
- tail *to print the last lines of a file*
- wc *count the words and lines in a text file*

cat command
viewing single file

A single argument, letter, is given to the cat command to process

```
/home/cis90/simben $ cat letter
Hello Mother!  Hello Father!

Here I am at Camp Granada.  Things are very entertaining,
and they say we'll have some fun when it stops raining.

All the counselors hate the waiters, and the lake has
alligators.  You remember Leonard Skinner?  He got
ptomaine poisoning last night after dinner.

Now I don't want this to scare you, but my bunk mate has
malaria.  You remember Jeffrey Hardy?  Their about to
organize a searching party.

Take me home, oh Mother, Father, take me home! I hate Granada.
Don't leave me out in the forest where I might get eaten
by a bear!  Take me home, I promise that I won't make noise,
or mess the house with other boys, oh please don't make me
stay -- I've been here one whole day.

Dearest Father, darling Mother, how's my precious little
brother?  I will come home if you miss me.  I will even
let Aunt Bertha hug and kiss me!

Wait a minute!  It's stopped hailing!  Guys are swimming!
Guys are sailing!  Playing baseball, gee that's better!
Mother, Father, kindly disregard this letter.

                                                Alan Sherman

/home/cis90/simben $
```

cat command

viewing multiple files

Multiple arguments, spellk and letter, are passed to the cat command to process

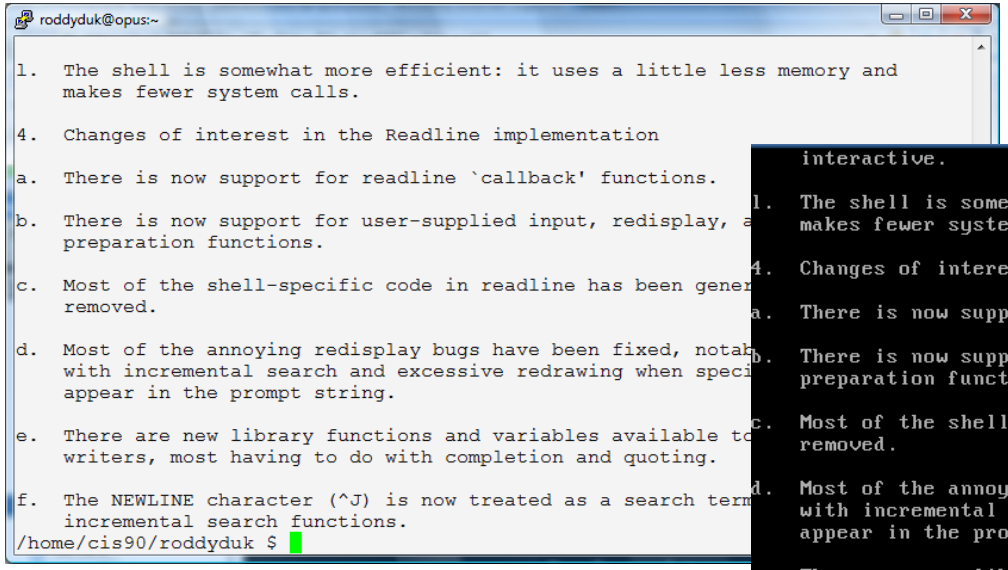
spellk

letter

```
/home/cis90/simben $ cat spellk letter
Spell Check


Eye halve a spelling chequer
It came with my pea sea
It plainly marques four my revue
< snipped >
Eye have run this poem threw it
I am shore your pleased two no
Its letter perfect awl the weigh
My chequer tolled me sew.

Hello Mother!  Hello Father!

Here I am at Camp Granada.  Things are very entertaining,
and they say we'll have some fun when it stops raining.
< snipped >
Wait a minute!  It's stopped hailing!  Guys are swimming!
Guys are sailing!  Playing baseball, gee that's better!
Mother, Father, kindly disregard this letter.

                                                  Alan Sherman
/home/cis90/simben $
```

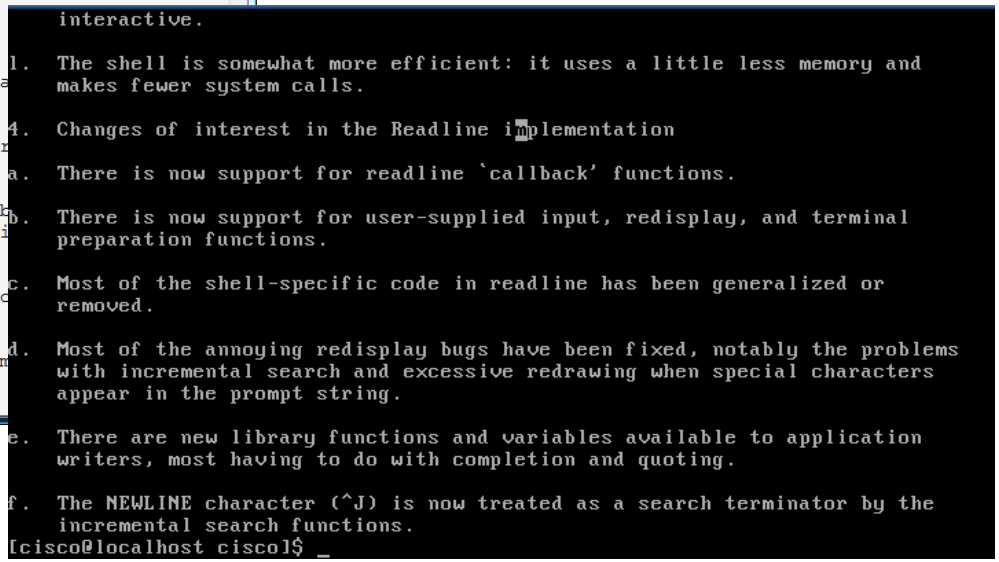
cat command
viewing long files

- Problem: if you **cat** really long files the text at the beginning is scrolled off and cannot be read.
- For example: **cat /usr/share/doc/bash-3.2/NEWS**



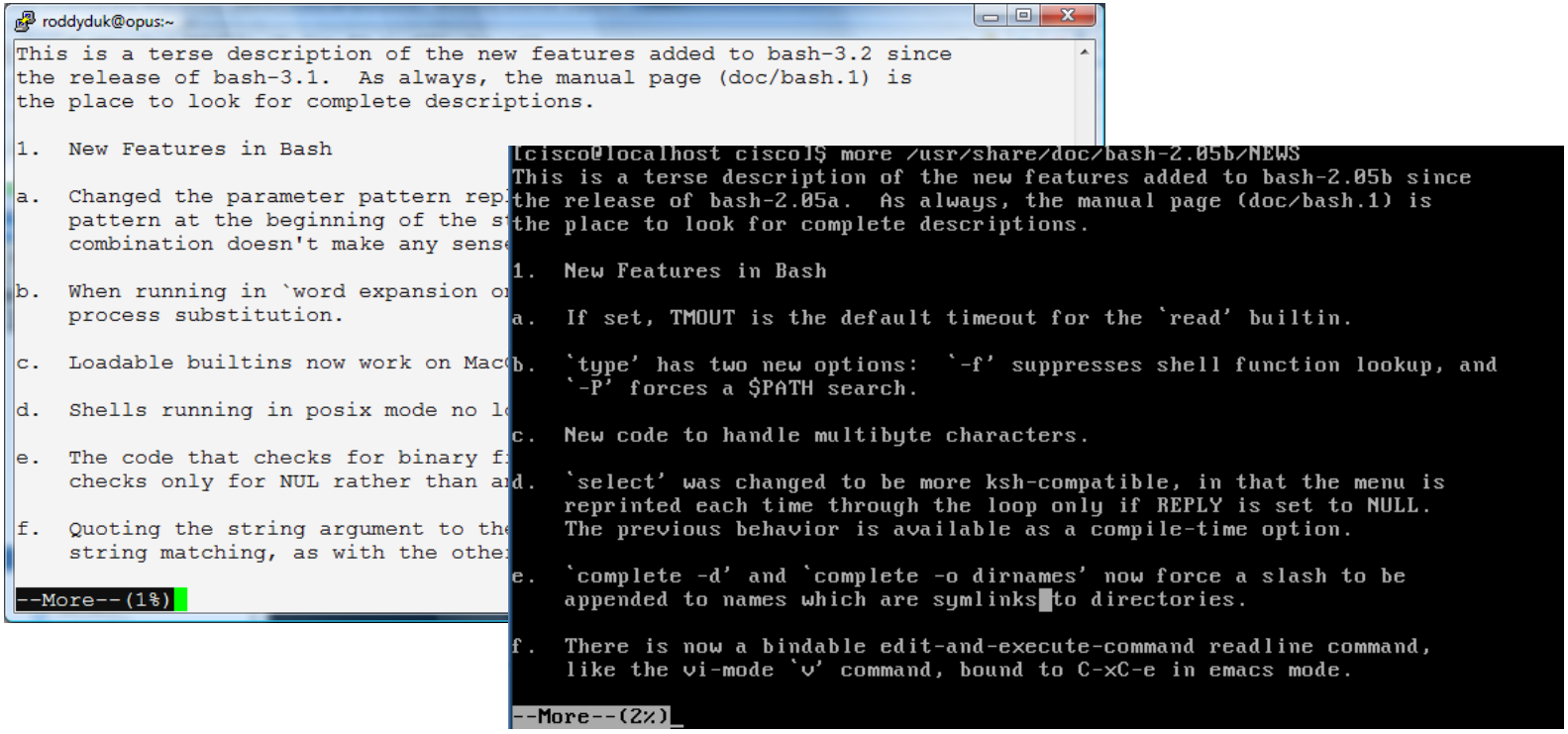
Terminal windows (like PuTTY) have scroll bars but the number of lines they buffer can be exceeded.

And virtual terminals have no scroll bars !



more command
viewing long files

- Use the **more** command for scolling through really long text files
- For example: **more /usr/share/doc/bash-3.2/NEWS**



Use the space key to page forward and q to quit

more command
viewing multiple files

- Use the **more** command can take multiple arguments

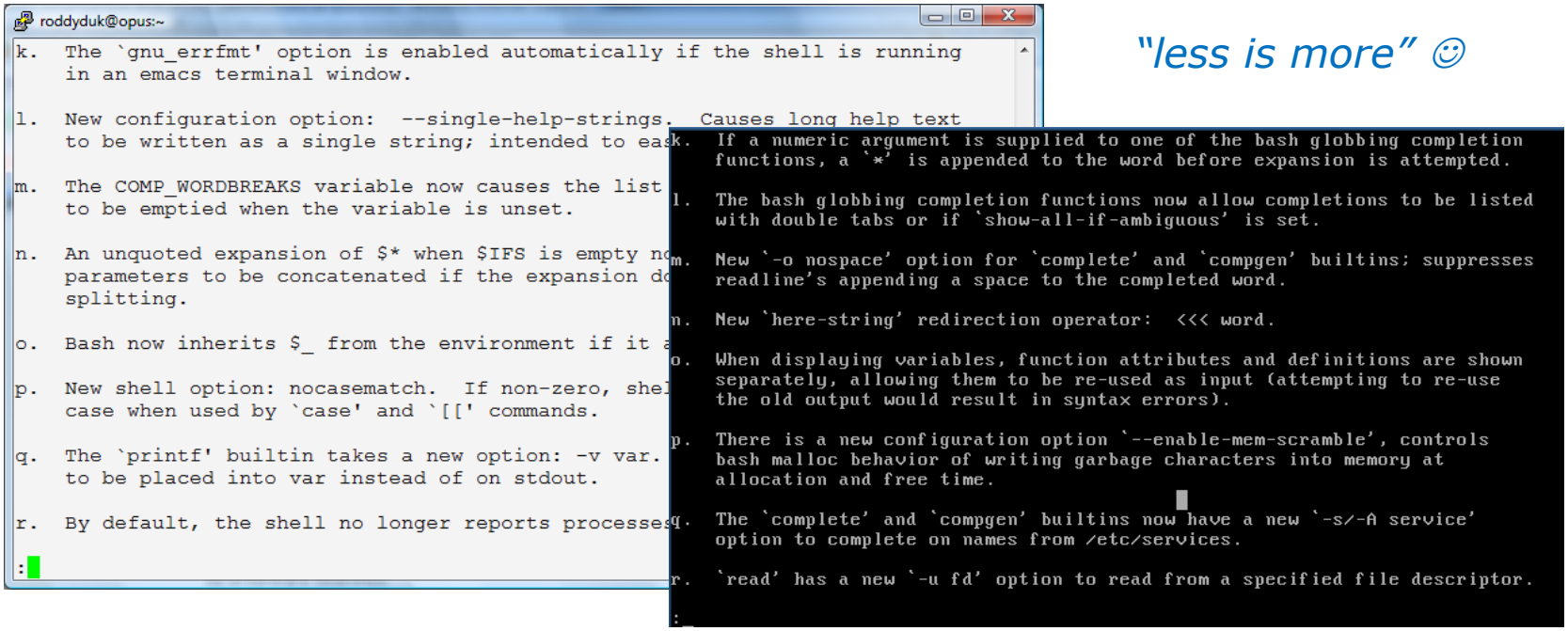
```
/home/cis90/simben $ more spellk letter
::::::::::::::
spellk
::::::::::::::
Spell Check


Eye halve a spelling chequer
It came with my pea sea
< snipped >
Its letter perfect awl the weigh
My chequer tolled me sew.

::::::::::::::
letter
::::::::::::::
Hello Mother!  Hello Father!
< snipped >
Guys are sailing!  Playing baseball, gee that's better!
Mother, Father, kindly disregard this letter.

                                                    Alan Sherman

/home/cis90/simben $
```

Notice with multiple files as arguments, each file has a header to separate it from the other files

less command
viewing long files

- Use the **less** command to scroll forward and backward through really long text files. (just like the man command works)

- For example: **less /usr/share/doc/bash-3.2/NEWS**



"less is more" ☺

*Use the pg up/dn and up/down arrows to move through text file. Use **q** to quit*
(See the man page for many more options like searching)

head command

view the first lines in a file

- Use the **head** command to show the first several lines of a file.
- Use the `–n <number>` option to control the number of lines printed.

A single argument, proposal1, is passed to the head command to process

```
/home/cis90/simben $ head proposal1
A Plan for the Improvement of English Spelling
   by Mark Twain
For example, in Year 1 that useless letter "c" would be dropped to be replased
either by "k" or "s", and likewise "x" would no longer be part of the alphabet.
The only kase in which "c" would be retained would be the "ch" formation, which
will be dealt with later. Year 2 might reform "w" spelling, so that "which" and
"one" would take the same konsonant, wile Year 3 might well abolish "y"
replasing it with "i" and Iear 4 might fiks the "g/j" anomali wonse and for all.
Jenerally, then, the improvement would kontinue iear bai iear with Iear 5 doing
awai with useless double konsonants, and Iears 6-12 or so modifaiing vowlz and
/home/cis90/simben $
```

One option, –n 3, and a single argument, proposal1, is given to the head command to process

```
/home/cis90/simben $ head -n 3 proposal1
A Plan for the Improvement of English Spelling
   by Mark Twain
For example, in Year 1 that useless letter "c" would be dropped to be replased
/home/cis90/simben $
```

head command

view the first lines of multiple files

```
/home/cis90/simben $ head -n 2 mission letter spellk log
==> mission <==
                        Mission * Purpose * Values

==> letter <==
Hello Mother!  Hello Father!

==> spellk <==
Spell Check

==> log <==
lab01 was submitted on Wed Feb  8 16:23:35 PST 2012
lab01 was submitted on Wed Feb  8 16:58:20 PST 2012
/home/cis90/simben $
```

One option , -n 2, and multiple arguments are passed to the head command to process

Note the small banner containing the filename which separates each file

tail command

view the last lines in a file

- Use the **tail** command to show the last several lines of a file.
- Use the –n <number> option to control the number of lines printed.

```
/home/cis90/simben $ tail mission
        environment which aids students in their pursuit of transfer,
        career preparation, personal fulfillment, job advancement, and
        retraining goals.

        Our core values are academic freedom, critical and independent
        thinking, and respect for all people and cultures. Our commitment
        is to encourage excellence, offer a balanced curriculum, promote
        teaching methods for diverse learning styles, and involve and
        enrich our community.

/home/cis90/simben $

/home/cis90/simben $ tail -n 3 mission
        teaching methods for diverse learning styles, and involve and
        enrich our community.

/home/cis90/simben $
```

wc command

count words and lines in a file

```
/home/cis90/simben $ wc letter
  28  182 1044 letter
```

#bytes
#words
#lines

```
/home/cis90/simben $ wc -l letter
28 letter
```

Use the –l option to count the number of words

```
/home/cis90/simben $ wc -w letter
182 letter
```

Use the –w option to count the number of words

```
/home/cis90/simben $ wc letter mission proposal1
  28  182 1044 letter
  18  107  759 mission
  16  196 1074 proposal1
  62  485 2877 total
```

The wc command can take multiple arguments

Class Exercise
Viewing Files

- Print the first 2 lines of mission, letter, spellk and log files

  head -n 2 mission letter spellk log

- Count the number of words in small_town

  wc –w small_town

Viewing Binary Files

Important commands for your toolbox:

- xxd *do a hex dump of a file*

binary data files

Binary files cannot be viewed with cat, more, less, head, tail, etc.

```
/home/cis90/simben $ cat /bin/uname
ELF04`I4(444444▒>▒>@▒( A  ▒▒HHH  P▒td▒644Q▒td/lib/ld-
linux.so.2GNU       (B▒`(*▒K▒▒G-▒▒>K▒▒▒y▒cg▒}Ti▒▒▒▒ w▒)
                                 ▒▒C52L▒/9=@▒x▒H_▒^fOI ▒
G▒<▒▒▒▒'▒▒6?▒▒▒w▒C*Y▒▒A▒▒$),K,∫▒"▒),K▒▒H▒.▒▒▒▒▒▒.▒▒.▒▒▒▒▒▒.▒▒▒▒
▒▒.▒▒▒▒▒/▒d8/▒▒▒▒▒</▒▒▒▒▒▒▒/▒▒▒▒▒▒▒/▒▒▒▒▒▒▒▒sii▒▒▒▒/▒▒ii▒▒▒w▒▒~▒▒w
▒▒~▒▒w▒▒~▒▒w▒▒~▒▒w▒▒ii
▒▒▒▒)▒*+, $(,08▒ <▒
```

< snipped >

```
PuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuT
TYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYP
uTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTT
YPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPuTTYPu
TTYPuTTYPuTTYPuTTY
/home/cis90/simben $
```

*Tip: Use **reset** command to fix terminal if it gets really "sick"*

xxd command

view hex dump of binary files

```
/home/cis90/simben $ xxd /bin/uname
0000000: 7f45 4c46 0101 0100 0000 0000 0000 0000  .ELF............
0000010: 0200 0300 0100 0000 308b 0408 3400 0000  ........0...4...
0000020: 6049 0000 0000 0000 3400 2000 0800 2800  `I......4. ...(.
0000030: 1f00 1e00 0600 0000 3400 0000 3480 0408  ........4...4...
0000040: 3480 0408 0001 0000 0001 0000 0500 0000  4...............
0000050: 0400 0000 0300 0000 3401 0000 3481 0408  ........4...4...
0000060: 3481 0408 1300 0000 1300 0000 0400 0000  4...............
0000070: 0100 0000 0100 0000 0000 0000 0080 0408  ................
< snipped >
0004df0: 0000 0000 0000 0000 d842 0000 6c05 0000  .........B..l...
0004e00: 0000 0000 0000 0000 0400 0000 0100 0000  ................
0004e10: 0100 0000 0300 0000 0000 0000 0000 0000  ................
0004e20: 4448 0000 1901 0000 0000 0000 0000 0000  DH..............
0004e30: 0100 0000 0000 0000                      ........
/home/cis90/simben $
```

Hexadecimal offsets into the file

Class Exercise

Where is the hostname command

type hostname

Then try to cat the hostname command:

cat /bin/hostname

Do a hex dump of the hostname command:

xxd /bin/hostname

Classifying Files

Important commands for your toolbox:

- file *to classify a file*

file command

Provides expanded information about files

- There are many different types of regular files:
  - Programs (binary)
  - Scripts (text)
  - Text files
  - Data files (binary)

- The **file** command attempts to classify files and give you more detailed information as to what type they are.

*Tip: Use the **file** command to determine is a file is a text file and can be viewed with **cat**, **more**, **less**, **tail** … etc commands.*

file command

Examples

Use the **file** command to identify the type of data in a file

```
/home/cis90/simben $ file letter
letter: ASCII English text
/home/cis90/simben $
```

*You can use **cat**, **more**, **head**, etc. safely on text files*

```
/home/cis90/simben $ file /bin/uname
/bin/uname: ELF 32-bit LSB executable, Intel 80386, version 1
(SYSV), for GNU/Linux 2.6.9, dynamically linked (uses shared
libs), for GNU/Linux 2.6.9, stripped
/home/cis90/simben $
```

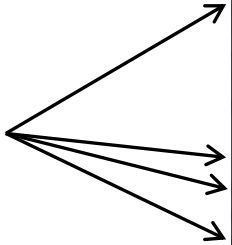
If it's not a text file then it could be a binary program or data file.

*Use **xxd** on binary files*

file command

Examples

*Use the **file** command to identify text files*



rsimms@opus:~/work/examples/filetypes

```
[rsimms@opus filetypes]$ file *
Adjective.frm:    MySQL table definition file Version 9
Adjective.MYD:    DBase 3 data file (33517822 records)
Adjective.MYI:    MySQL MISAM compressed data file Version 1
backup.tar:       POSIX tar archive
bash_profile:     ASCII English text
console:          character special (5/1)
cprog:            ELF 32-bit LSB executable, Intel 80386, version
 1 (SYSV), for GNU/Linux 2.2.5, dynamically linked (uses shared l
ibs), for GNU/Linux 2.2.5, not stripped
go-cprog:         symbolic link to `cprog'
letter:           ASCII English text
mbox:             ASCII mail text
rich-260x216.jpg: JPEG image data, JFIF standard 1.02
runit:            Bourne shell script text executable
sda:              block special (8/0)
travel:           directory
[rsimms@opus filetypes]$
```

Class Activity

Classify the following files in your home directory:

- mbox
- letter
- Poems
- timecal

```
/home/cis90/simben $ file mbox letter Poems timecal
mbox:    ASCII mail text, with very long lines
letter:  ASCII English text
Poems:   directory
timecal: shell archive or script for antique kernel text
/home/cis90/simben $
```

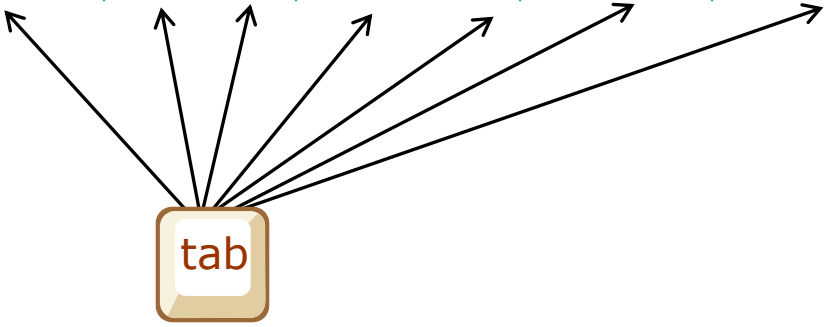
Shell tips

bash shell tip

tab completes

- It can be tedious typing in long pathnames.
- Since bash knows the names of the files you only have to type just enough characters to uniquely specify a name and then the tab key can be pressed to complete them.
- Example: the black characters were typed by the user, the green ones were typed by bash:

```
ls /home/cis90/simben/Poems/Shakespeare/
```

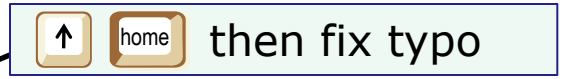


bash shell tip

command history and editing

- It can be tedious re-typing a long command to fix a typo.
- Since bash knows the commands you have previously entered, just use the up and down arrows to re-type a previous command.
- When the command you want appears, use the home, right or left arrow keys to go where you want to make the correction. New text can be inserted and old text deleted or backspaced over.
- Example: The ls command was mis-typed as la:

```
/home/cis90/simmsben $ la /home/cis90/simmsben/Poems/Shakespeare/
-bash: la: command not found
```




```
/home/cis90/simmsben $ ls /home/cis90/simmsben/Poems/Shakespeare/
sonnet1   sonnet11  sonnet17  sonnet26  sonnet35  sonnet5  sonnet9
sonnet10  sonnet15  sonnet2   sonnet3   sonnet4   sonnet7
/home/cis90/simmsben $
```

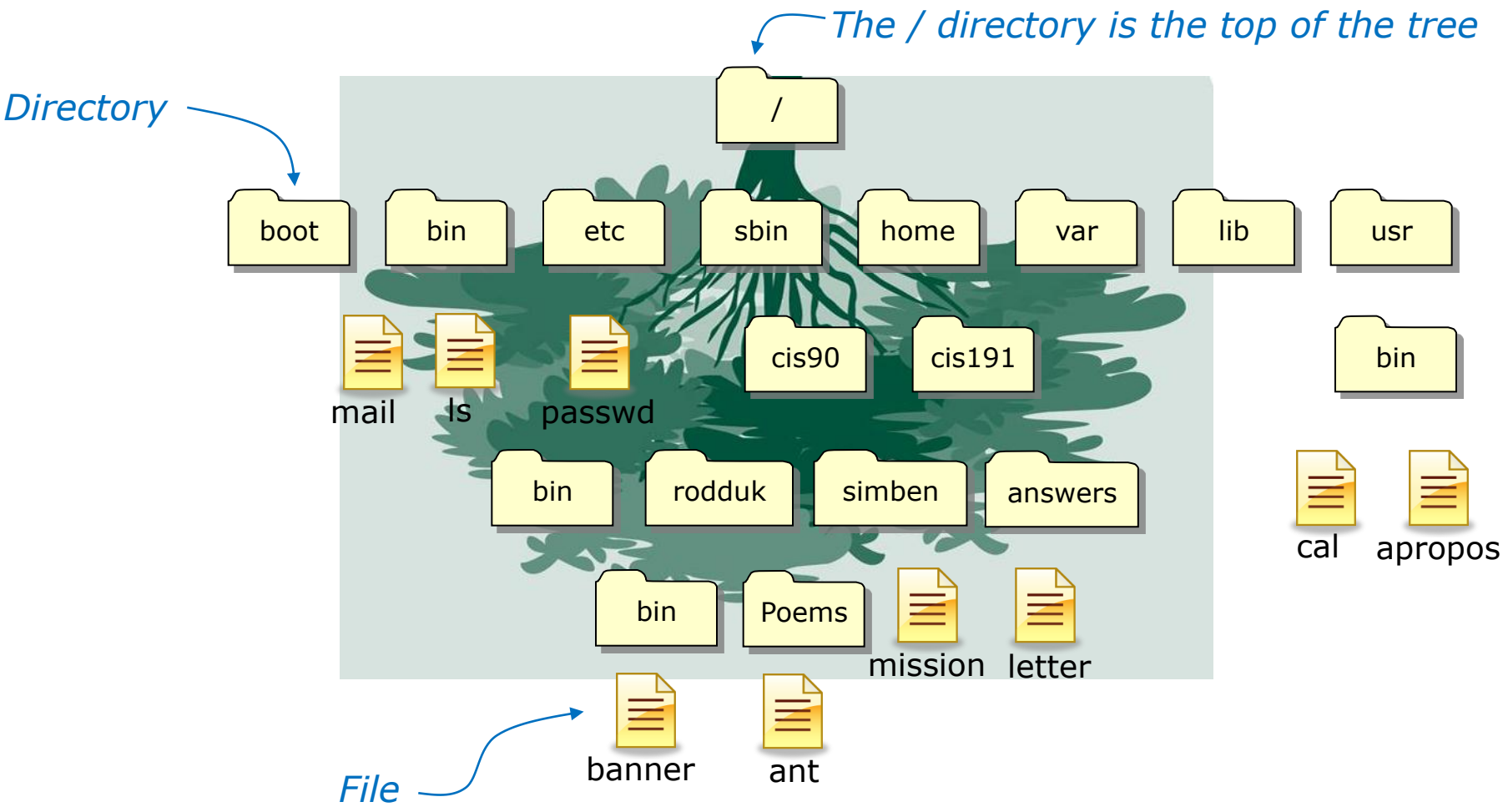
The UNIX Directory Hierarchy

UNIX File Tree

/ = root of the tree

UNIX File Tree

/ = root of the tree

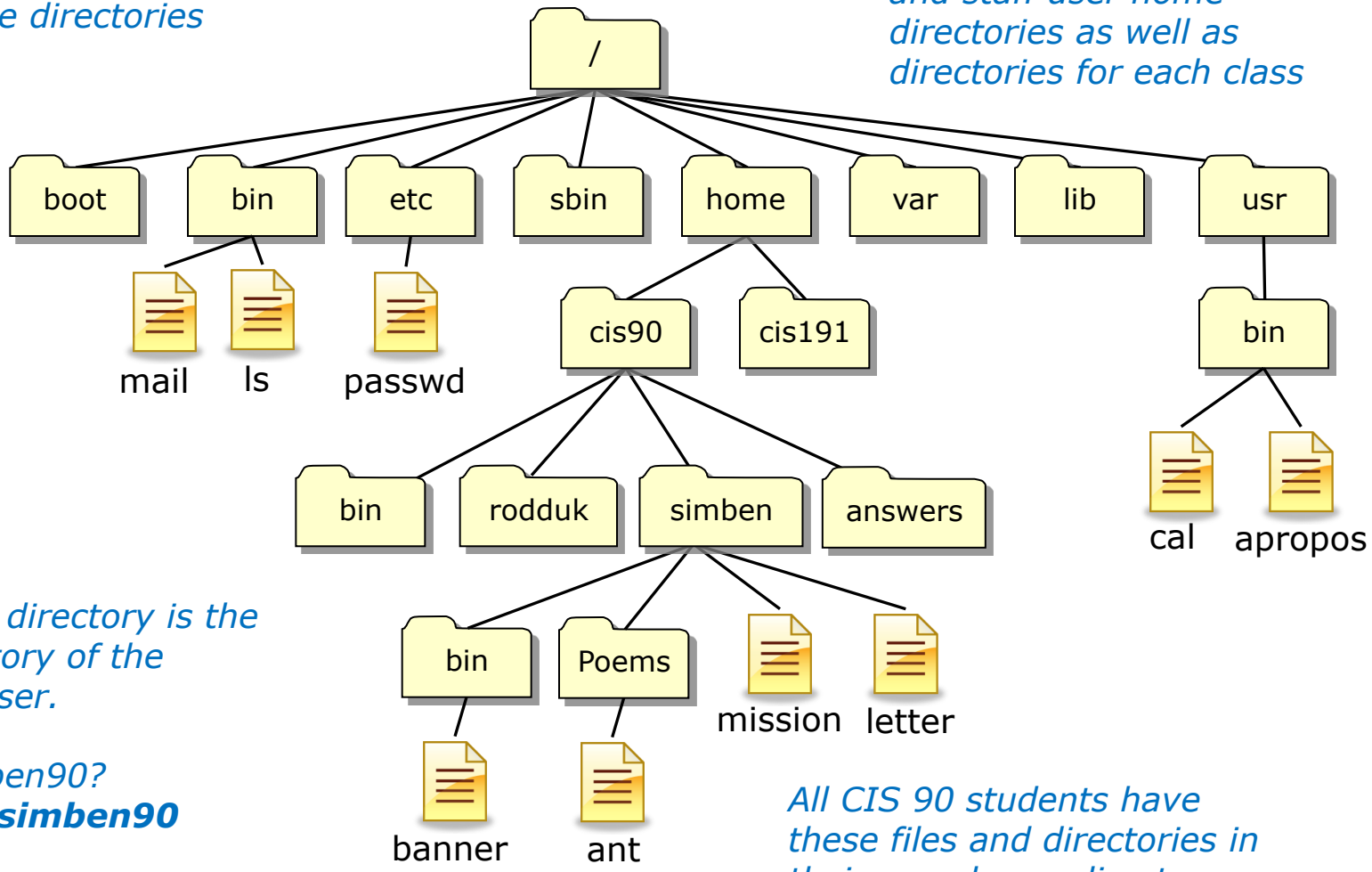


UNIX File Tree

/ = root of the tree

Directories may contain files or more directories

On Opus, the directory named "home" has faculty and staff user home directories as well as directories for each class

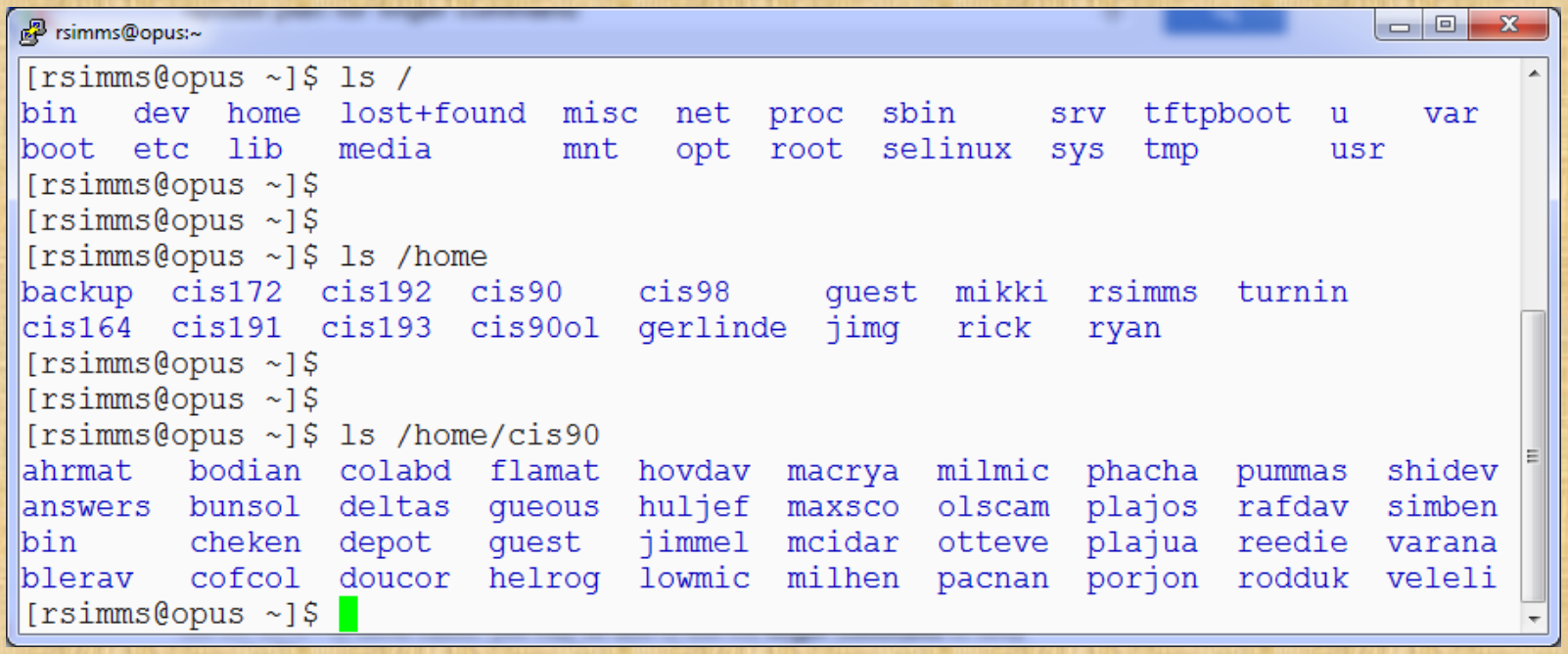


The simben directory is the home directory of the simben90 user.

Who is simben90?
*Use **finger simben90***

All CIS 90 students have these files and directories in their own home directory

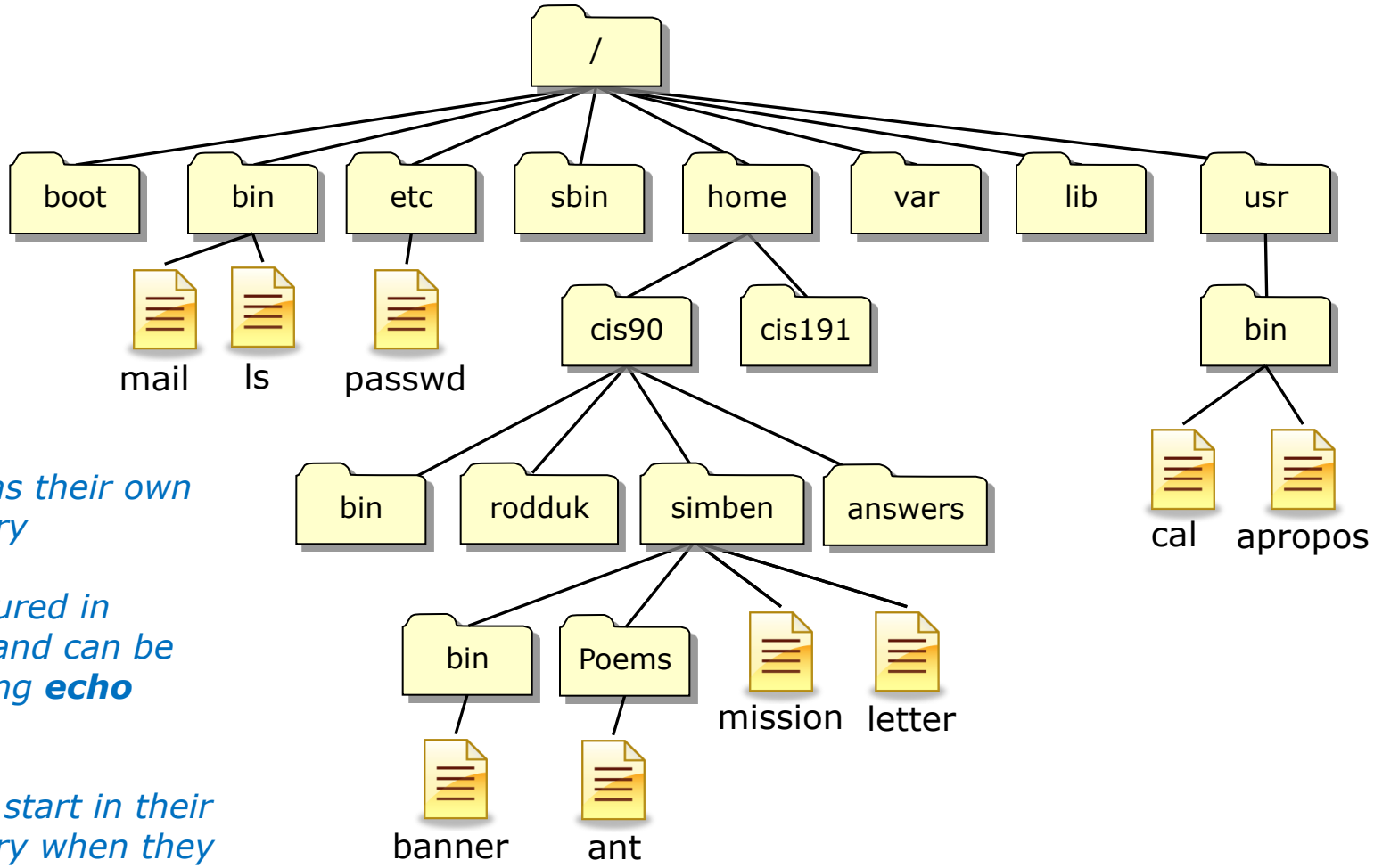
Class Activity



1) List the top level **/** directory
2) Locate **/etc**, ***/home***, ***/bin***, ***/sbin*** *and* ***/usr*** directories in the output
3) List the contents of the directory named */home*
4) *Locate the instructors home directory in the output*
5) List the contents of the directory named ***/home/cis90***
6) *Locate your own home directory in the output*

UNIX File Tree

/ = root of the tree

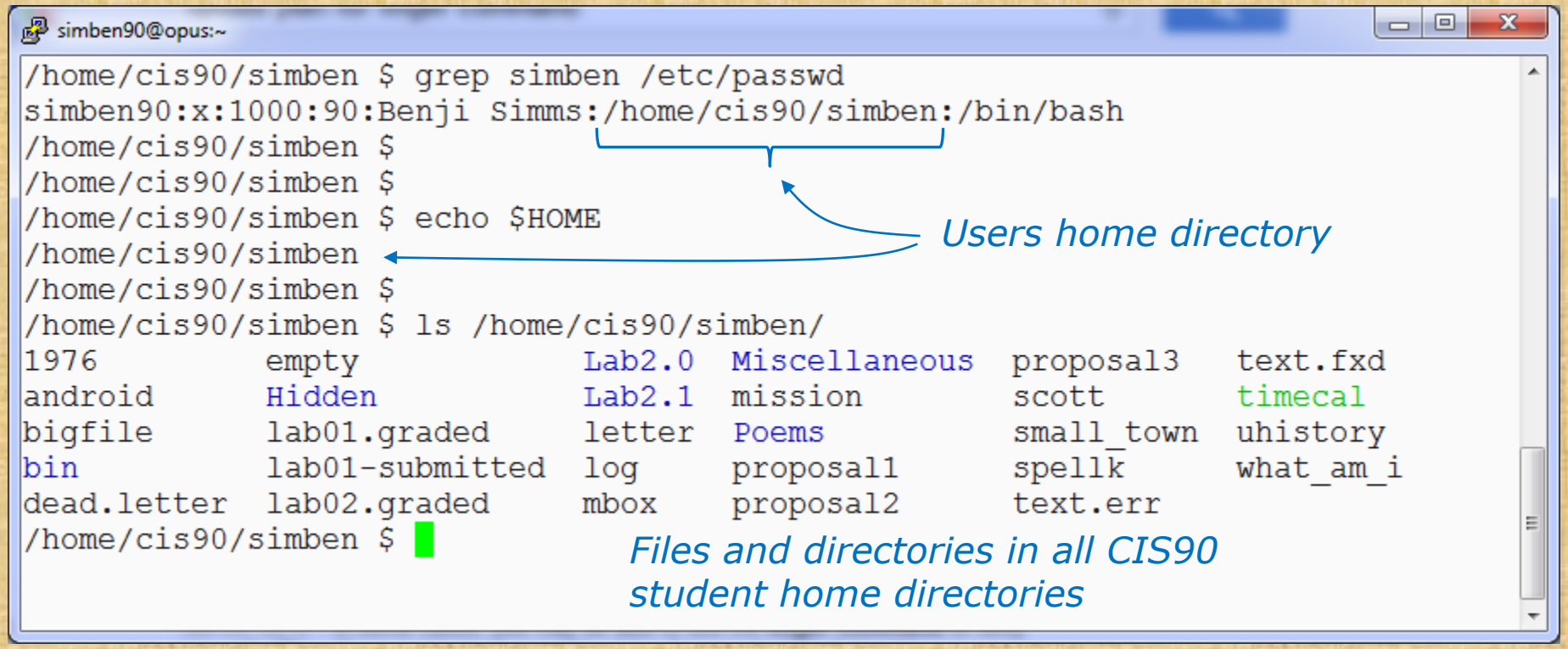


Every user has their own home directory

This is configured in /etc/passwd and can be displayed using ***echo $HOME***

Users always start in their home directory when they login

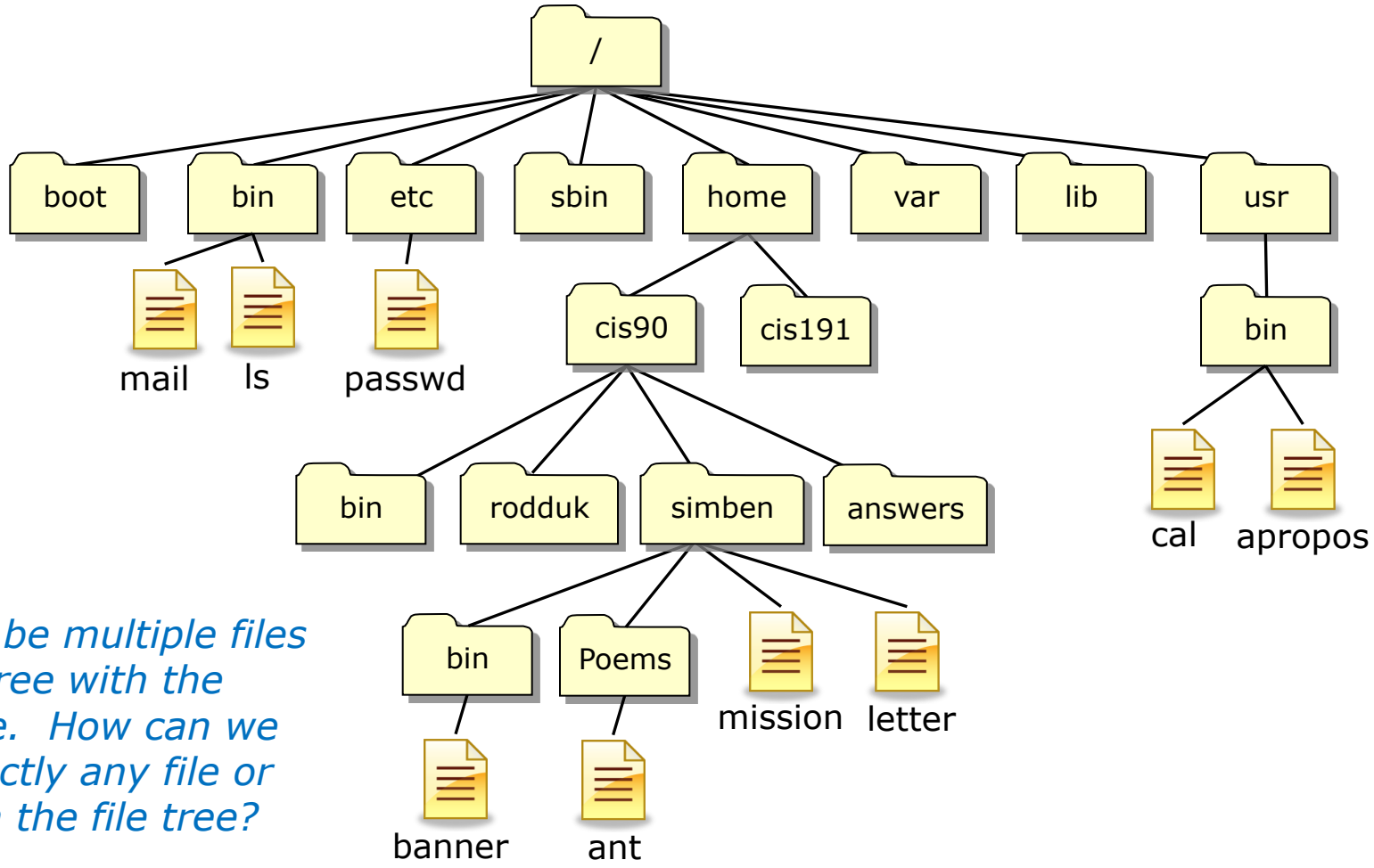
Class Activity



1) Find your entry in /etc/passwd and locate your home directory
2) Show the contents of the HOME variable
3) List the contents of your home directory

UNIX File Tree

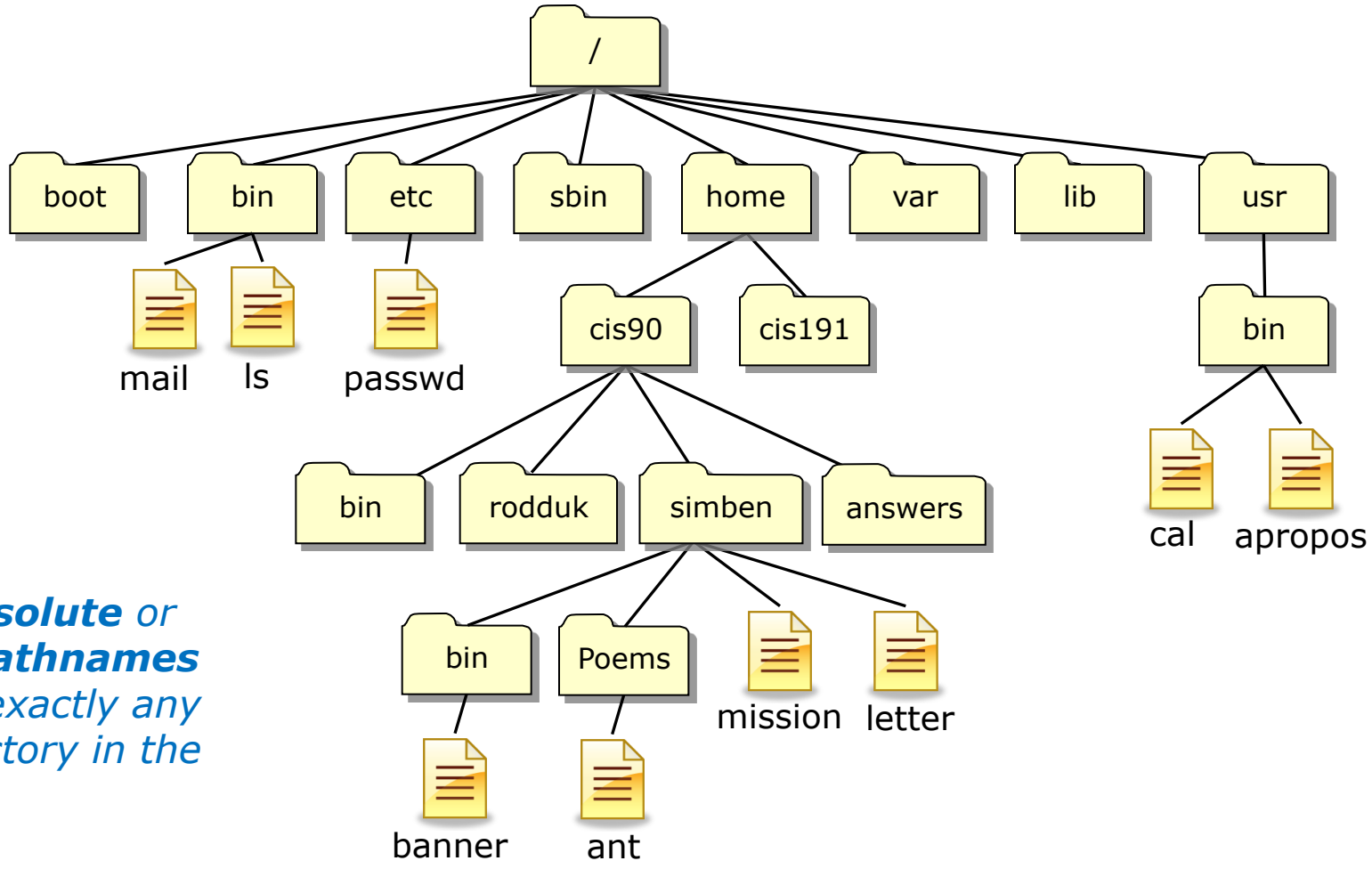
/ = root of the tree



There may be multiple files in the file tree with the same name. How can we specify exactly any file or directory in the file tree?

UNIX File Tree

/ = root of the tree



*We use **absolute** or **relative pathnames** to specify exactly any file or directory in the tree.*

Pathnames

What the heck are they?

A pathname is a precise way to specify any file or directory in the file tree.

- An **absolute pathname** specifies the path from the top of the tree to the target directory or file.
- A **relative pathname** specifies the path from your current location to the target directory or file.

Understanding pathnames is critical because they are used as arguments to all commands that deal with files and directories.

Absolute Pathnames

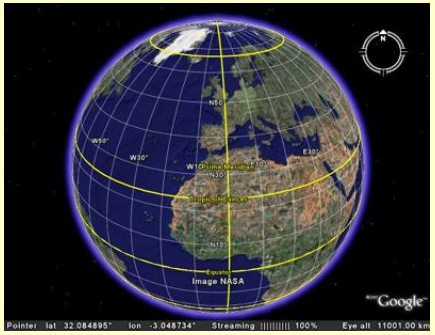
*An **absolute pathname** specifies the path from the top of the tree to the target directory or file.*

Examples:

/home/cis90/simben/Poems/ant	*(file)*
/boot	*(directory)*
/usr/bin/cal	*(file)*
/home/cis90/bin/	*(directory)*
/bin/mail	*(file)*

Notice they all start with the /

http://www.engineeringtoolbox.com/



Note, latitude and longitude is an example of specifying a location in an absolute fashion based on the equator and prime meridian

Aptos, CA
Latitude: 36-58'52" N
Longitude: 121-52'28" W

Absolute Pathnames

Using absolute pathnames as command arguments

*An **absolute pathname** specifies the path from the top of the tree to the target directory or file.*

Examples of absolute pathnames used as command arguments:

ls /bin /sbin /usr/bin /usr/sbin

file /usr/bin/cal

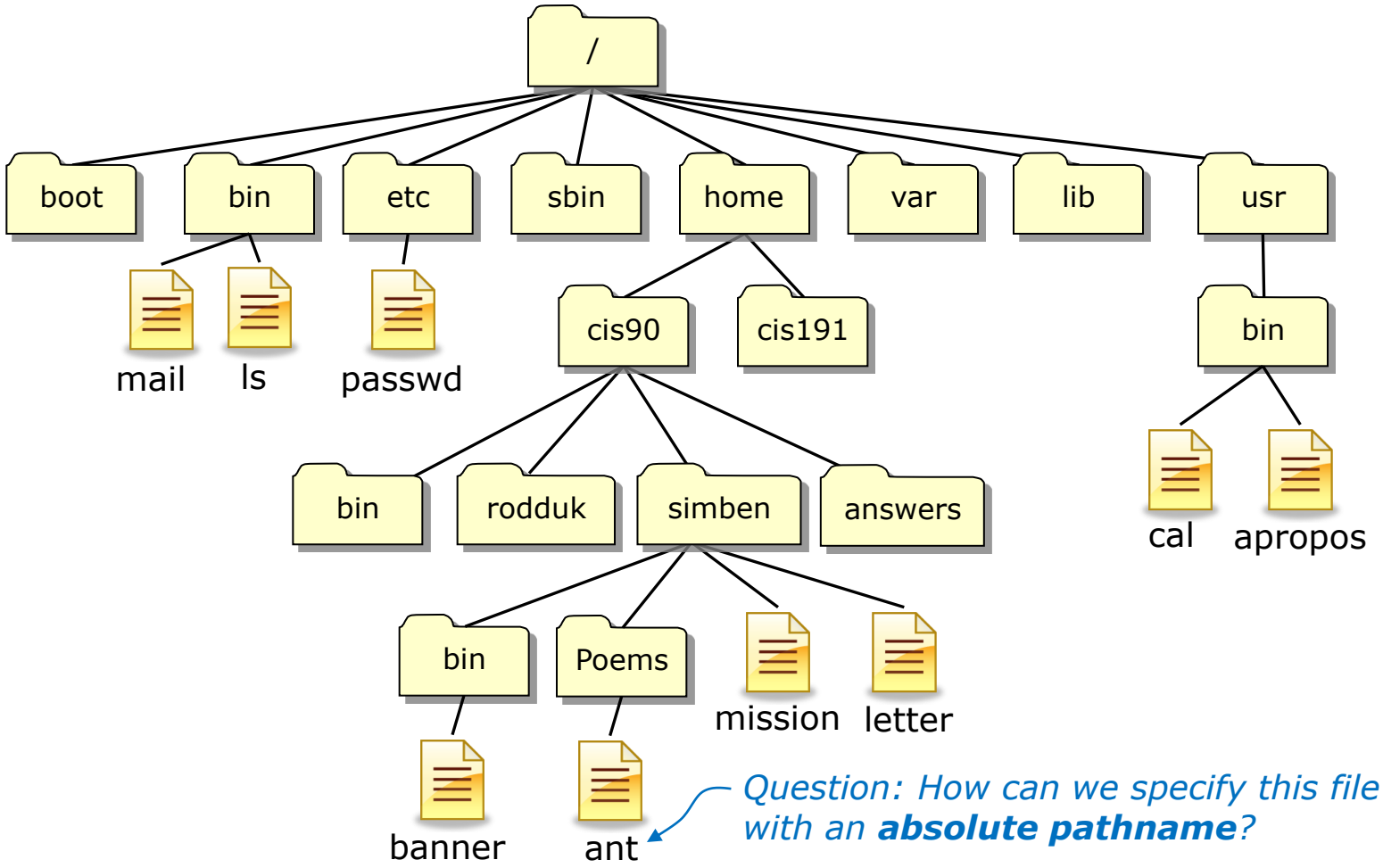
cd /home/cis90/simben/Poems/Shakespeare

ls -l /bin/mail

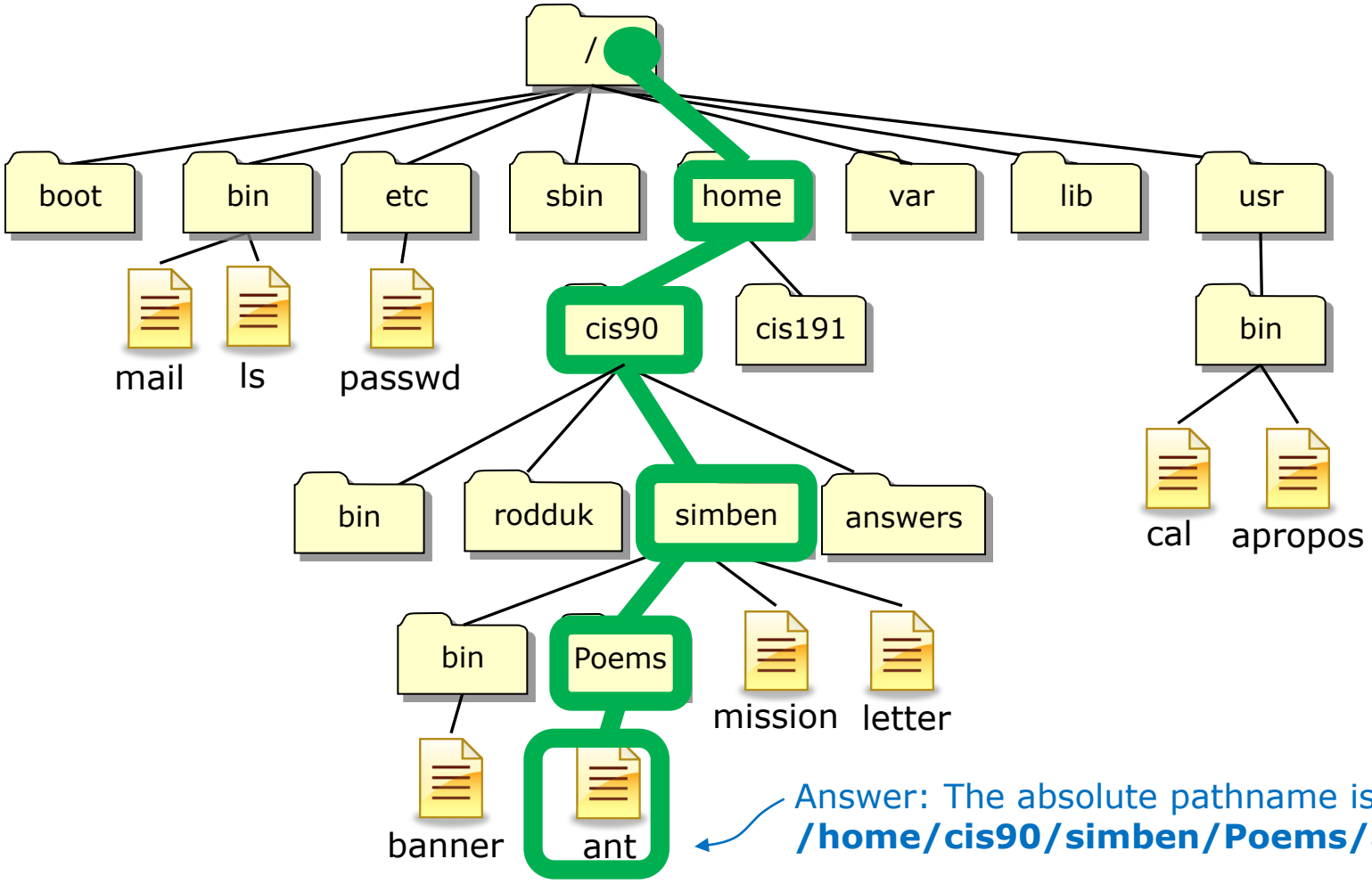
cp /etc/passwd /home/cis90/simben/misc

Notice all pathnames start with the /

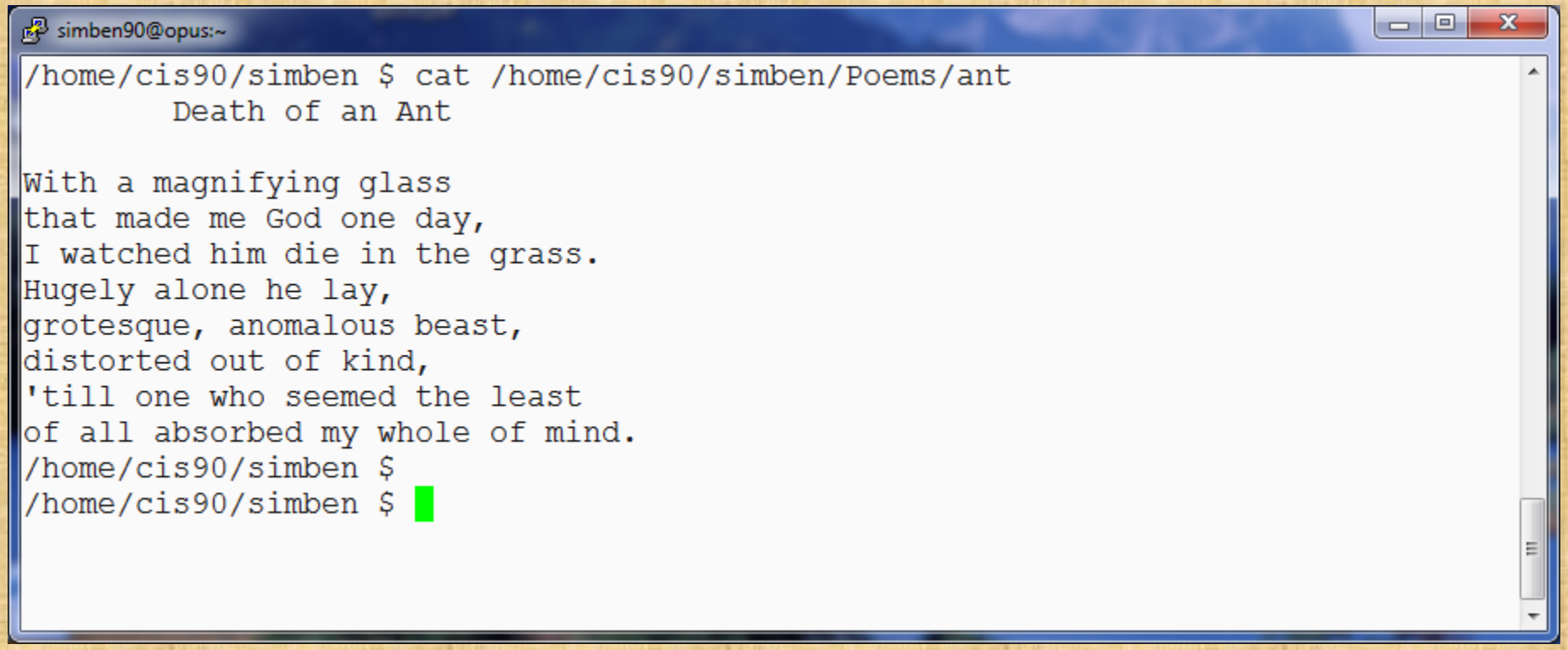
An **absolute pathname** specifies the path from the top of the tree to the target directory or file



An **absolute pathname** specifies the path from the top of the tree to the target directory or file



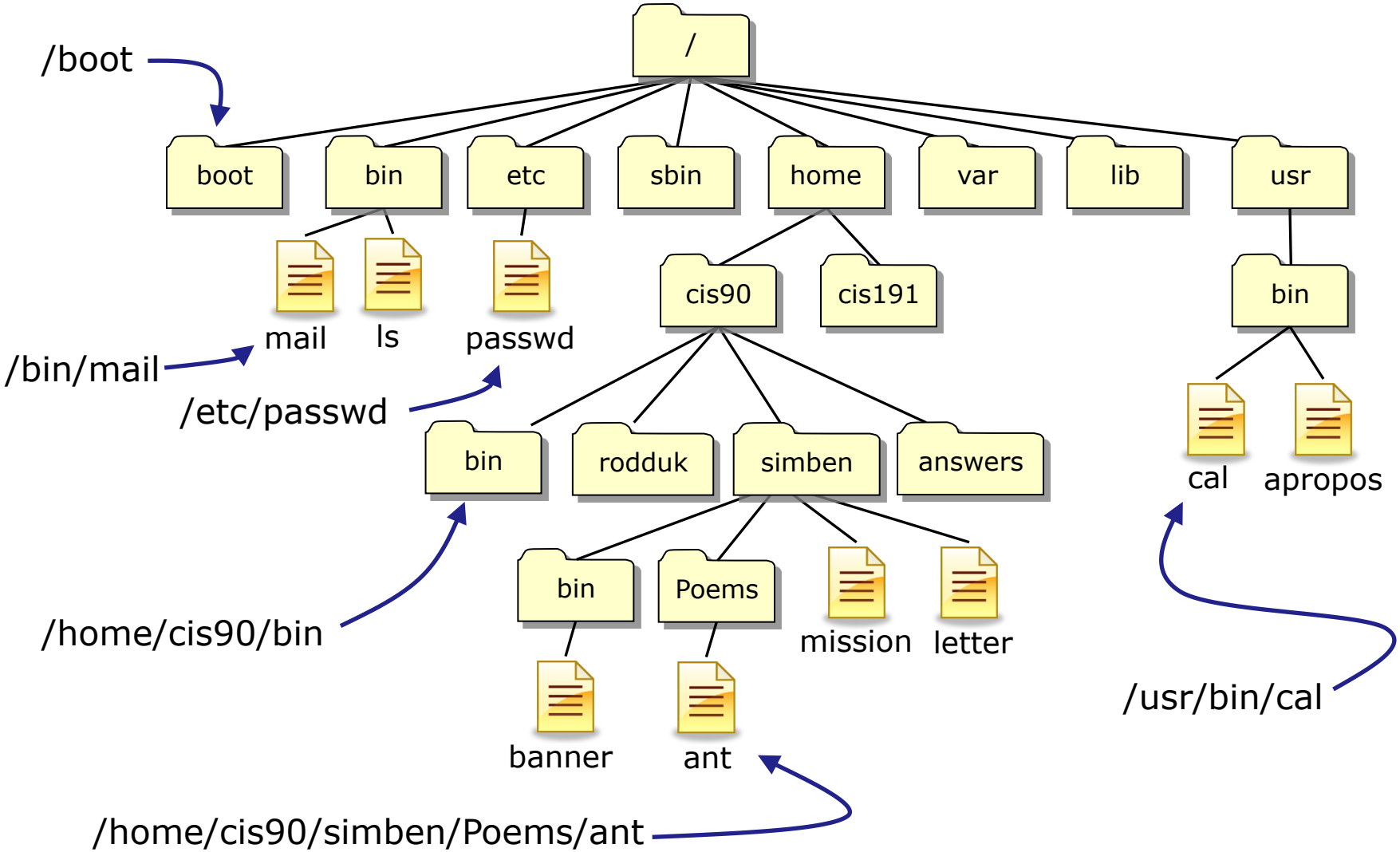
Class Activity



1) List the *ant* file belonging to simben90 using an absolute pathname
2) List your *ant* using an absolute pathname

Absolute Pathnames

These are all absolute pathname examples



Relative Pathnames

*A **relative pathname** specifies the path from your current location to the target directory or file.*

Examples:

ant	*(file)*
Poems/Shakespeare/sonnet5	*(file)*
../mission	*(file)*
../bin/	*(directory)*
../../../boot/vmlinuz-2.6.18-164.el5	*(file)*

Note that relative pathnames do NOT start with a /

Note, Google Maps is a way of specifying a location in a relative fashion based with step-by-step instructions from your current location

Relative Pathnames

Using relative pathnames as command arguments

*A **relative pathname** specifies the path from your current location to the target directory or file.*

Examples of using relative pathnames as command arguments:

ls -l ant

file ../../../../bin/mail

cd Poems/Blake

ls -l ../bin/check3

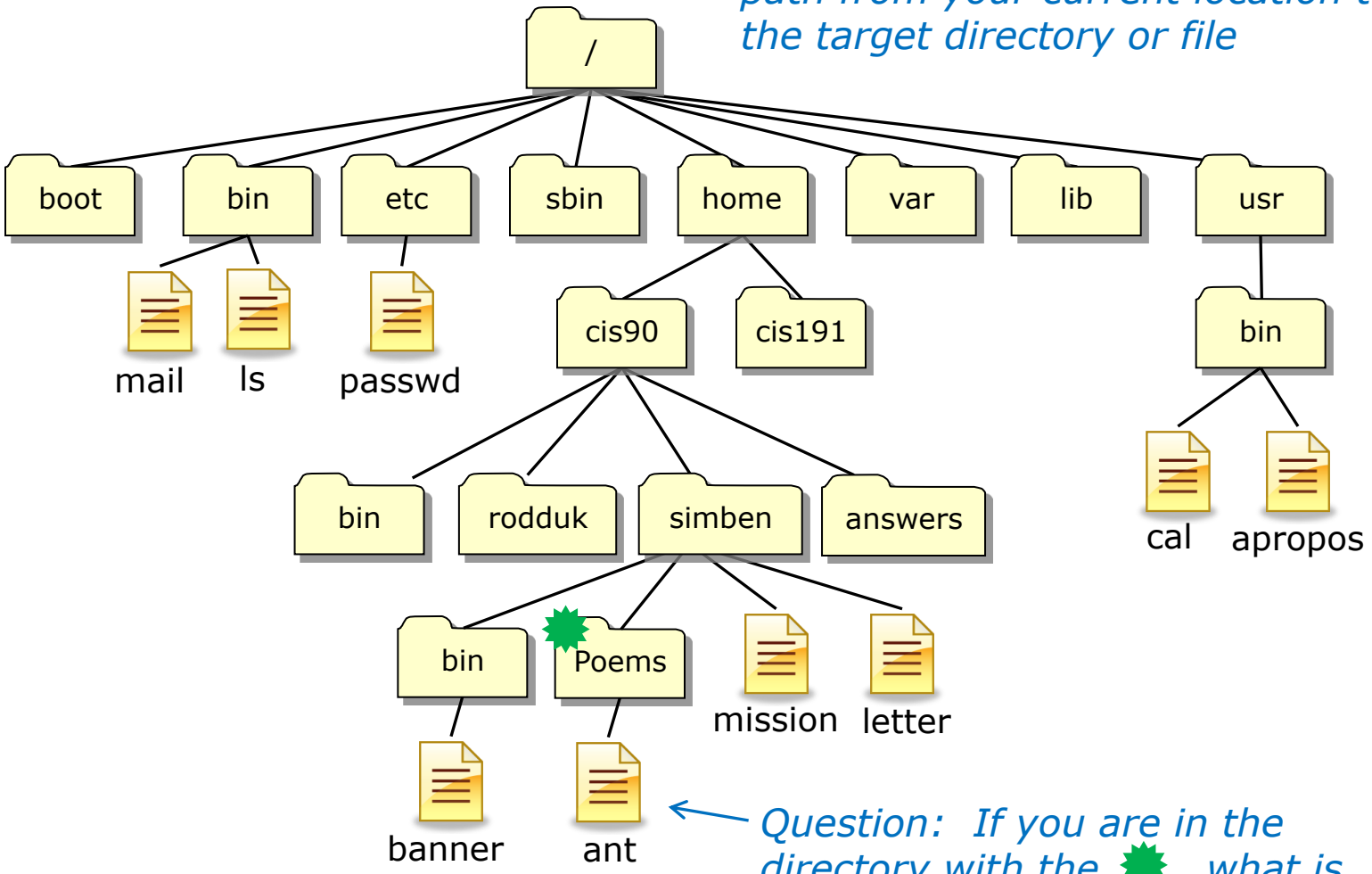
file Poems/Shakespeare/sonnet4

cd Poems/Shakespeare

Notice that these pathnames do NOT start with the /

Relative Pathname Example

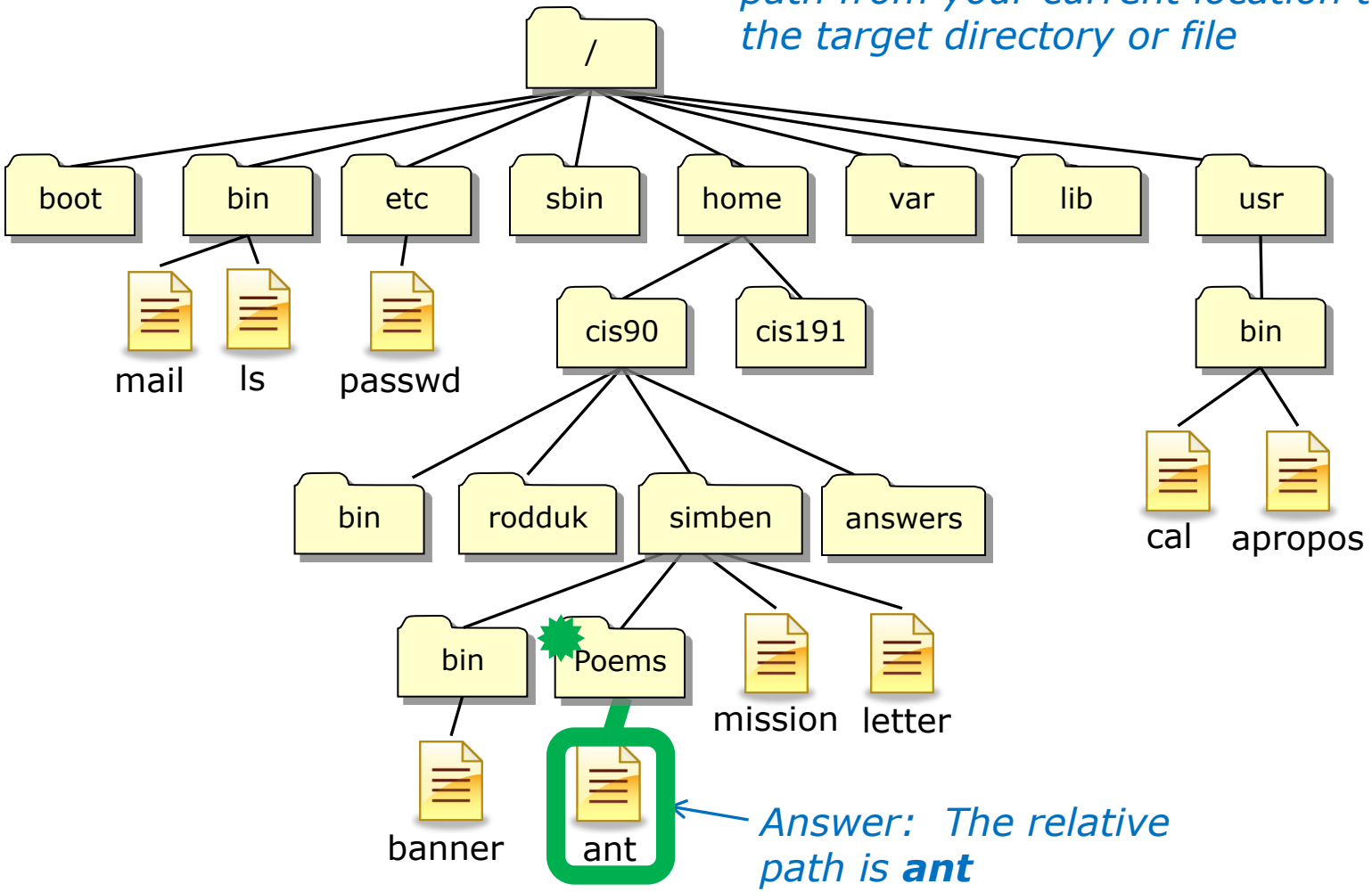
*A **relative pathname** specifies the path from your current location to the target directory or file*



Question: If you are in the directory with the , what is the relative path to this file?

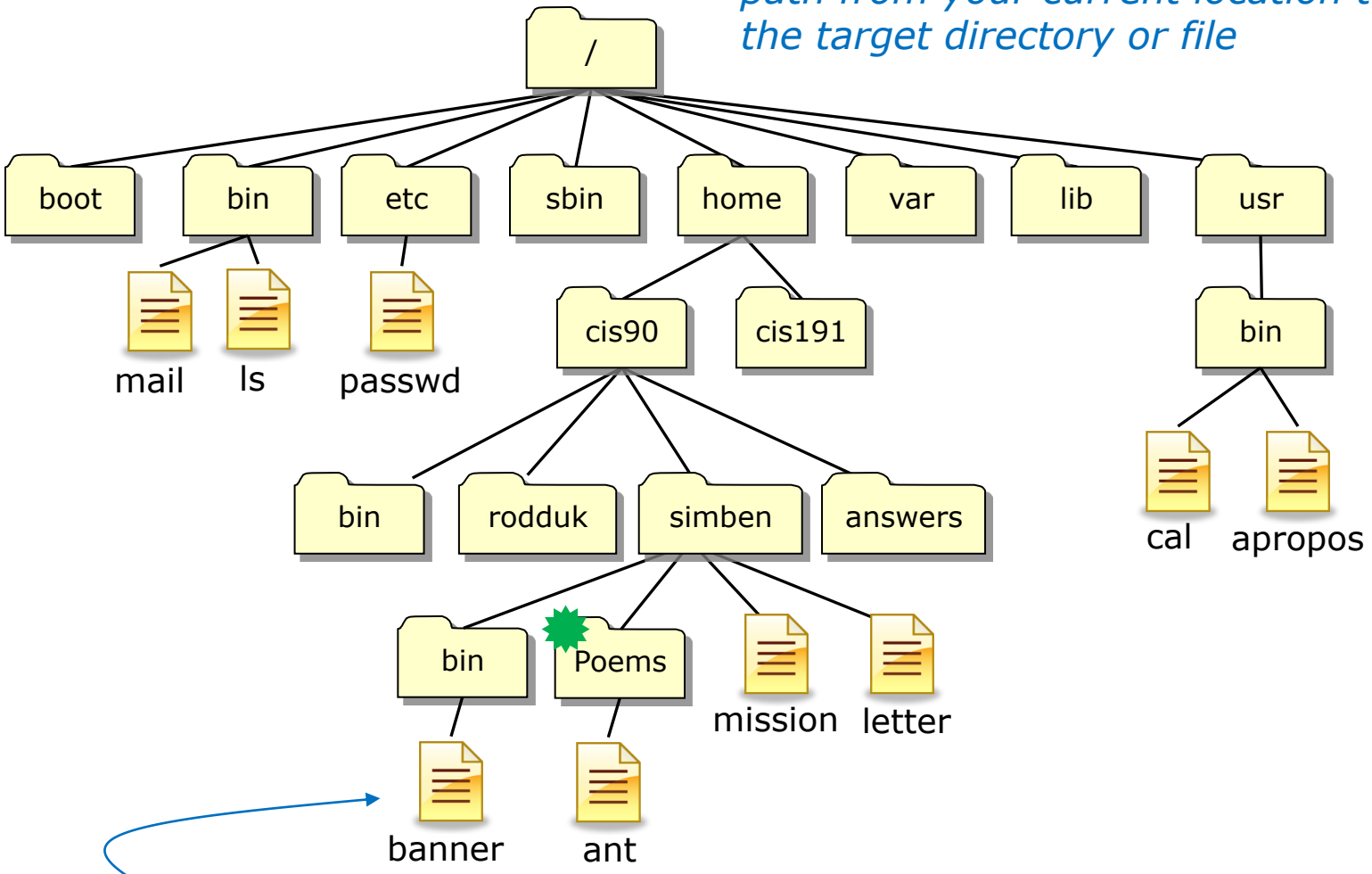
Relative Pathname Example

*A **relative pathname** specifies the path from your current location to the target directory or file*



Relative Pathname Example

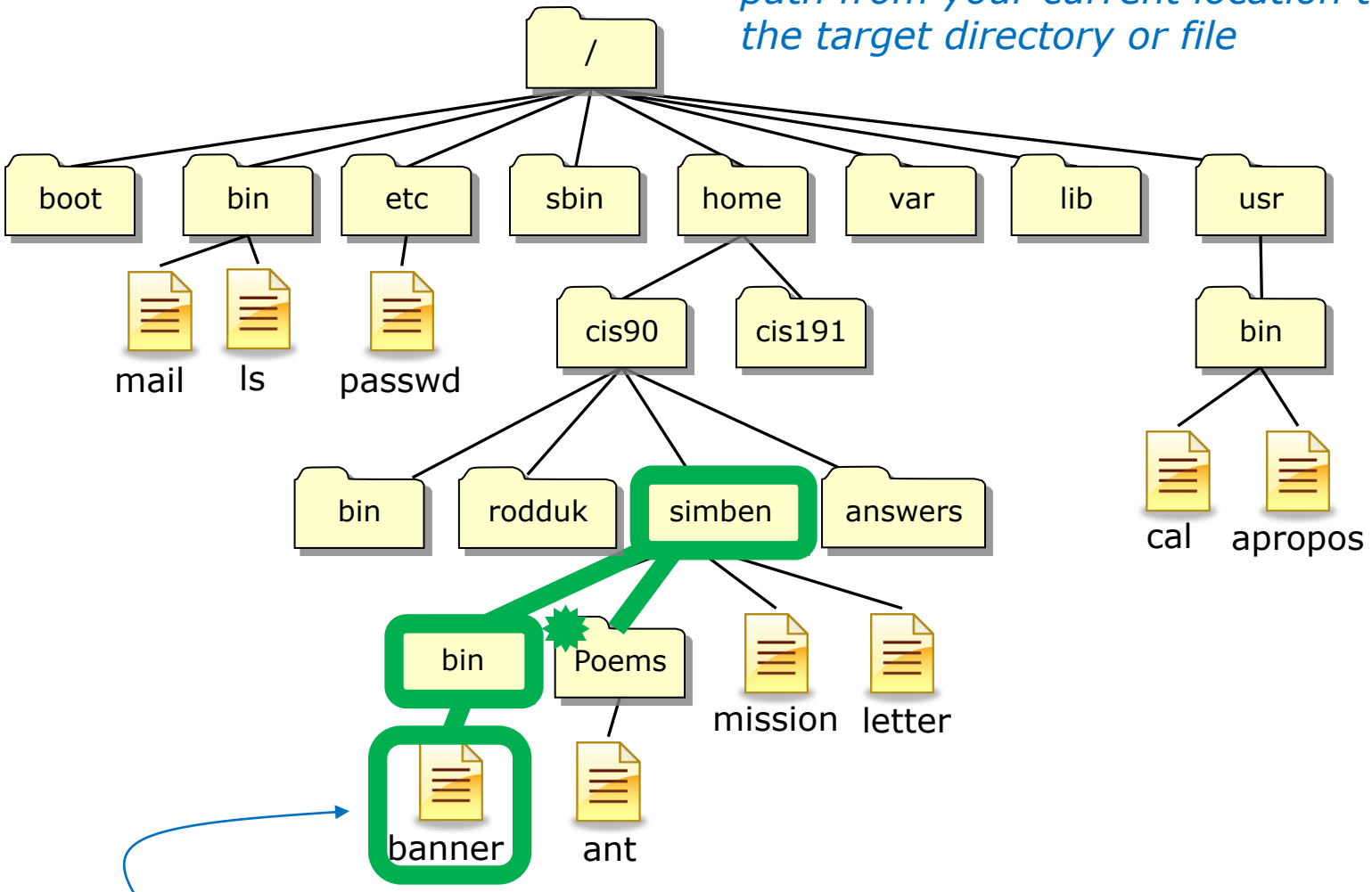
*A **relative pathname** specifies the path from your current location to the target directory or file*



Question: If you are in the directory with the ✹, what is the relative path to this file?

Relative Pathname Example

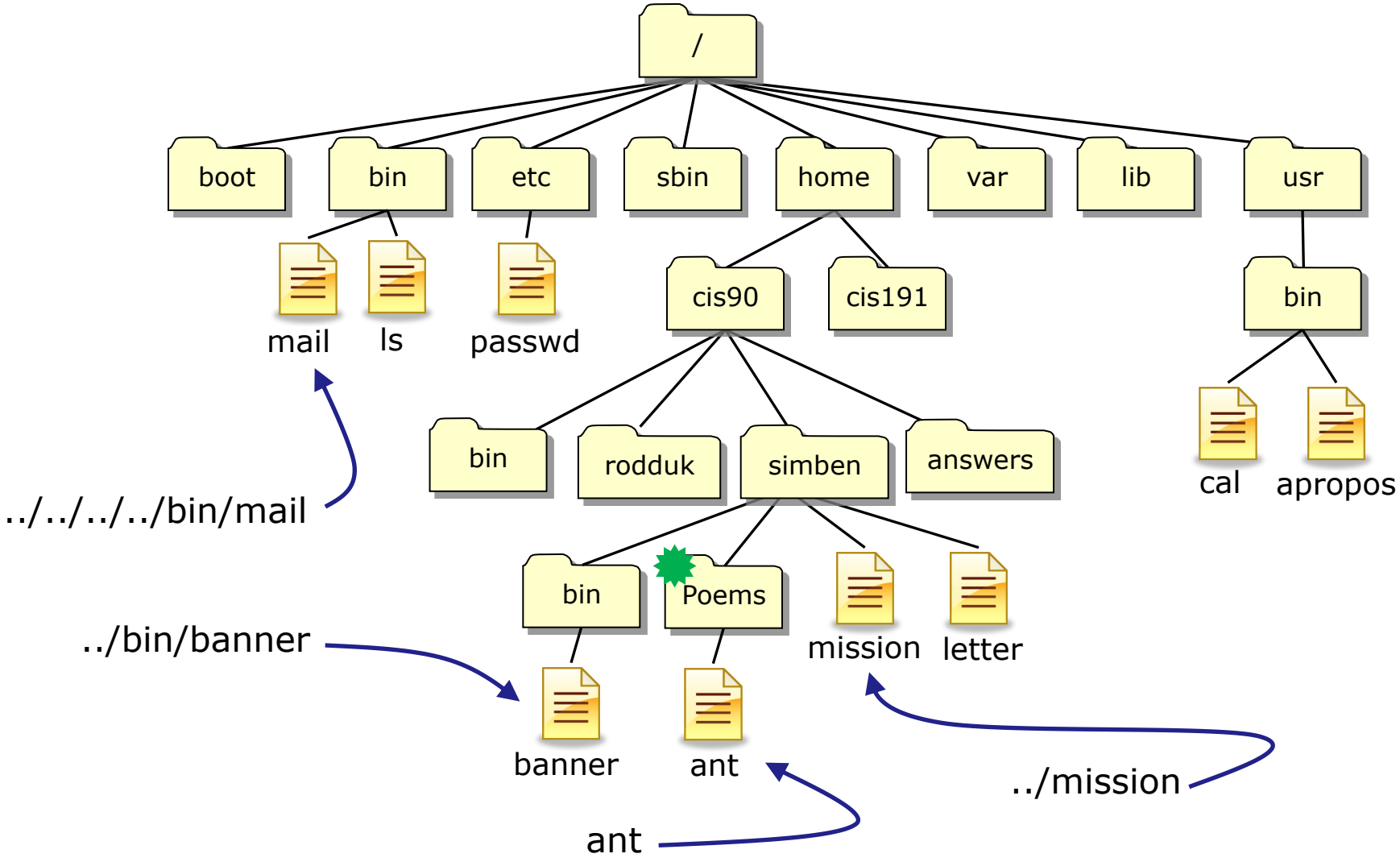
*A **relative pathname** specifies the path from your current location to the target directory or file*



*Answer: The relative path to this file is **../bin/banner***

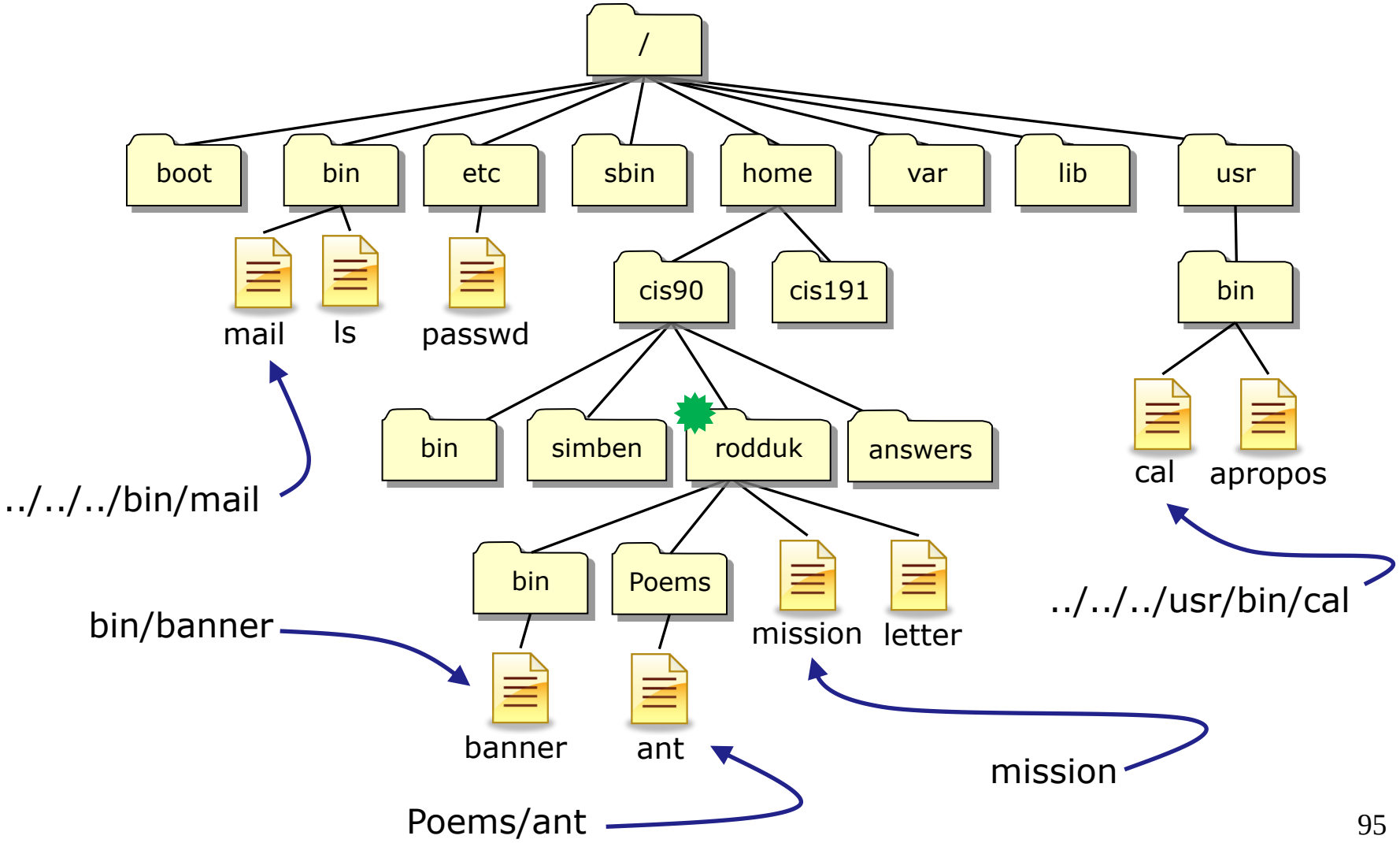
Relative Pathnames

Names that start relative to the current working directory (✹)



Relative Pathnames

Names that start relative to the current working directory (✹)



Class Exercise

From your home directory:

- List the /etc/passwd/ file using a relative pathname

  ls ../../../etc/passwd

- List the /etc/passwd file using a absolute pathname

  ls /etc/passwd

  Sometimes it's easier to specify a filename using an absolute pathname

- List the letter file using a relative pathname

  ls letter

  Sometimes it's easier to specify a filename using a relative pathname

- List the letter file using an absolute pathname

  ls /home/cis90/simben/letter

user your home directory instead

Heads up on a future test question

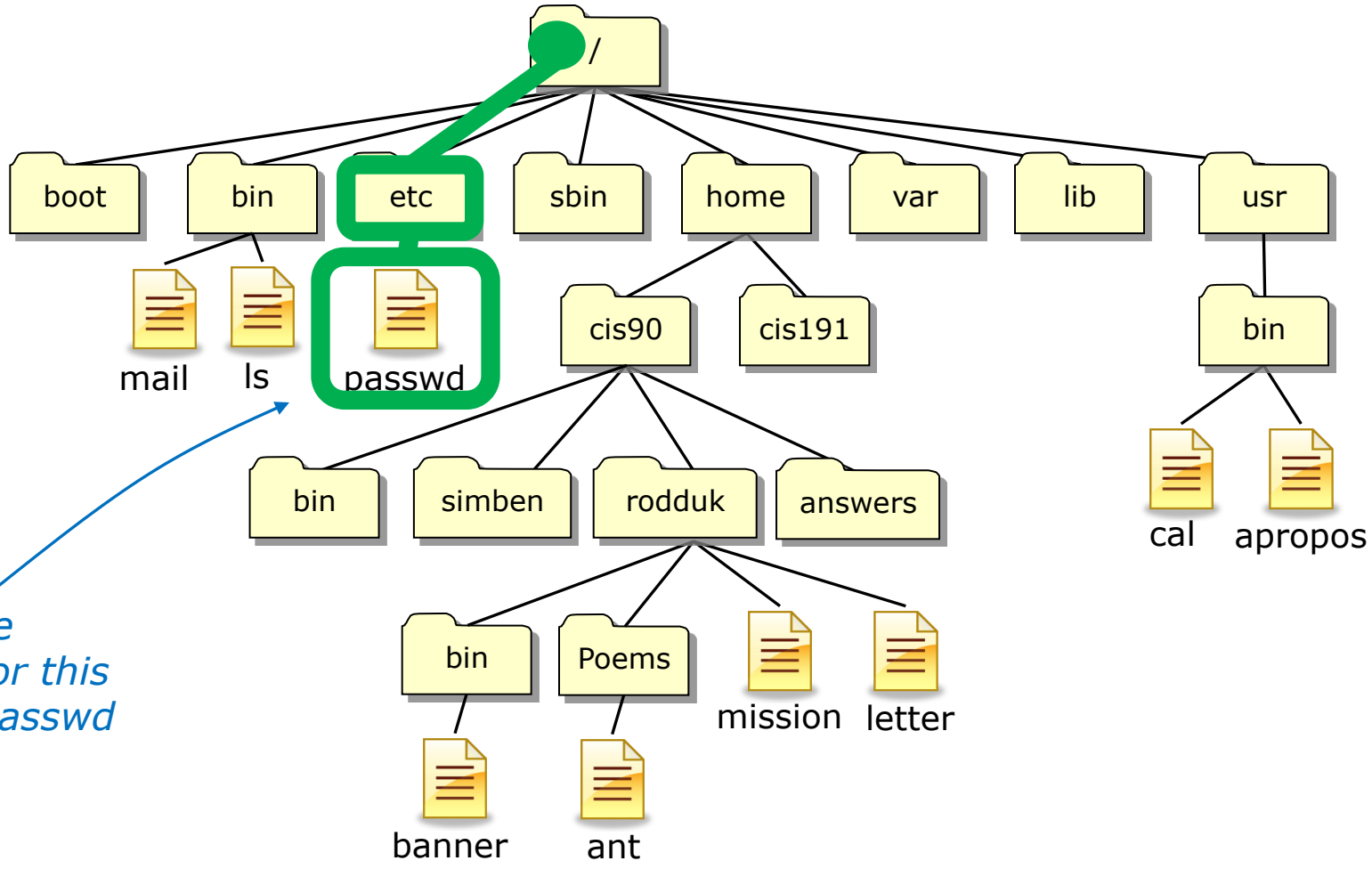
Question: What is the absolute pathname of /etc/passwd?

Answer: /etc/passwd

What is the color of Washington's white horse?

UNIX File Tree

/ = root of the tree



The absolute pathname for this file is /etc/passwd

/

.

. .

~

More on Directories

/ by itself is the root or "/" directory, the top of the tree, not to be confused with the root user's home directory (/root)

/ at the beginning of a pathname indicates an absolute path

/ at the end of a filename indicates it is a directory

.. is shorthand for the absolute path to your current **parent** directory

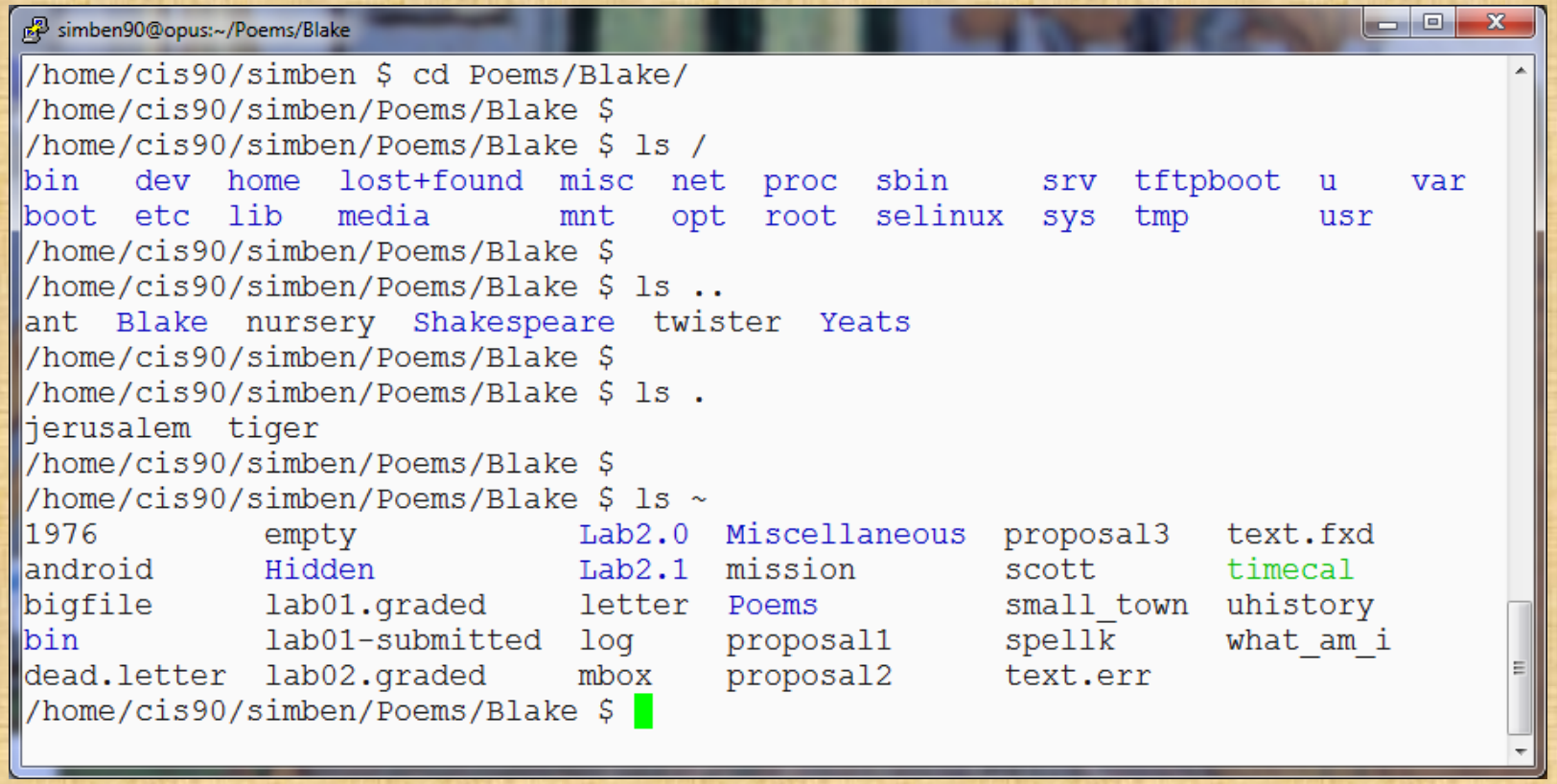
. is shorthand for the absolute path to your current directory = "here"

~ is shorthand for the absolute path to your home directory

. and .. are hidden files, more on hidden files later

Class Activity

1. Change to your Poems/Blake directory using a relative pathname
2. List the directories in the / directory using an absolute pathname
3. List the directories in your current parent directory using ..
4. List the directories in your current directory using .
5. List the file in your home directory using ~



UNIX File Hierarchy

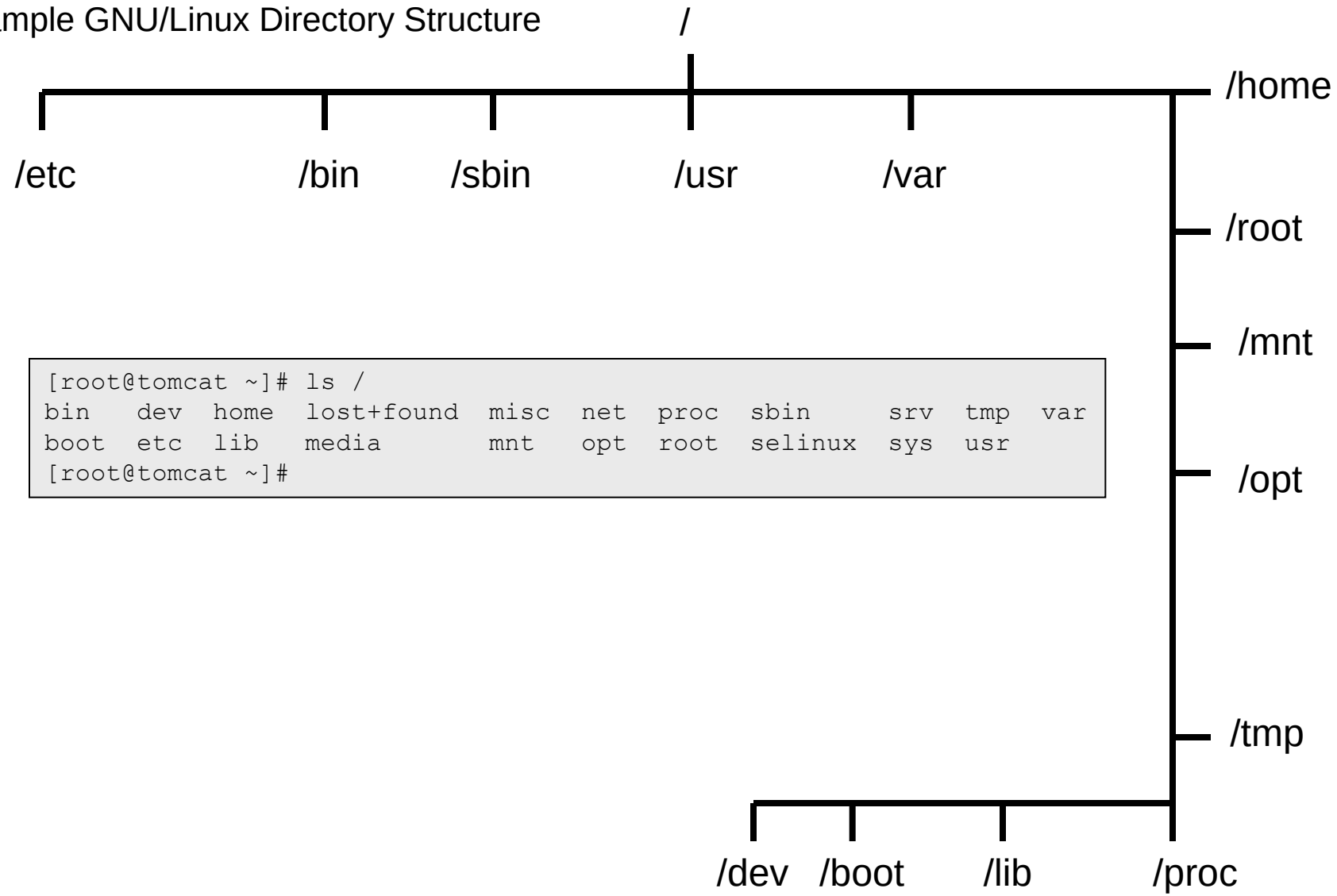
- /
  - /bin
  - /boot
  - /dev
  - /etc
  - /home
  - /lib
  - /lost+found
  - /mnt
  - /opt
  - /proc
  - /root
  - /sbin
  - /tmp
  - /usr

The UNIX/Linux File System Hierarchy

There are standard top level directories in every version of UNIX/Linux

Directory	Contents
/bin	binary files forming the commands and shells used by the system administrator and users
/boot	files used during the initial bootup process including the kernel
/dev	device files, like terminals and drives for connected hardware
/etc	system configuration files
/home	individual directories owned by each user
/lib	shared libraries needed to boot the system and run the commands in the root filesystem (i.e. commands in /bin and /sbin)
/lost+found	recovered files that were corrupted by power failures or system crashes
/mnt	mount points for floppies, cds, or other file systems
/opt	add-on software packages and/or commercial applications
/proc	kernel level process information
/root	home directory for the root user
/sbin	system administration commands reserved for the superuser (root)
/tmp	temporary files that are deleted when the system is rebooted or started
/usr	program files and related files for use by all users
/var	log files, print spool files, and mail queues

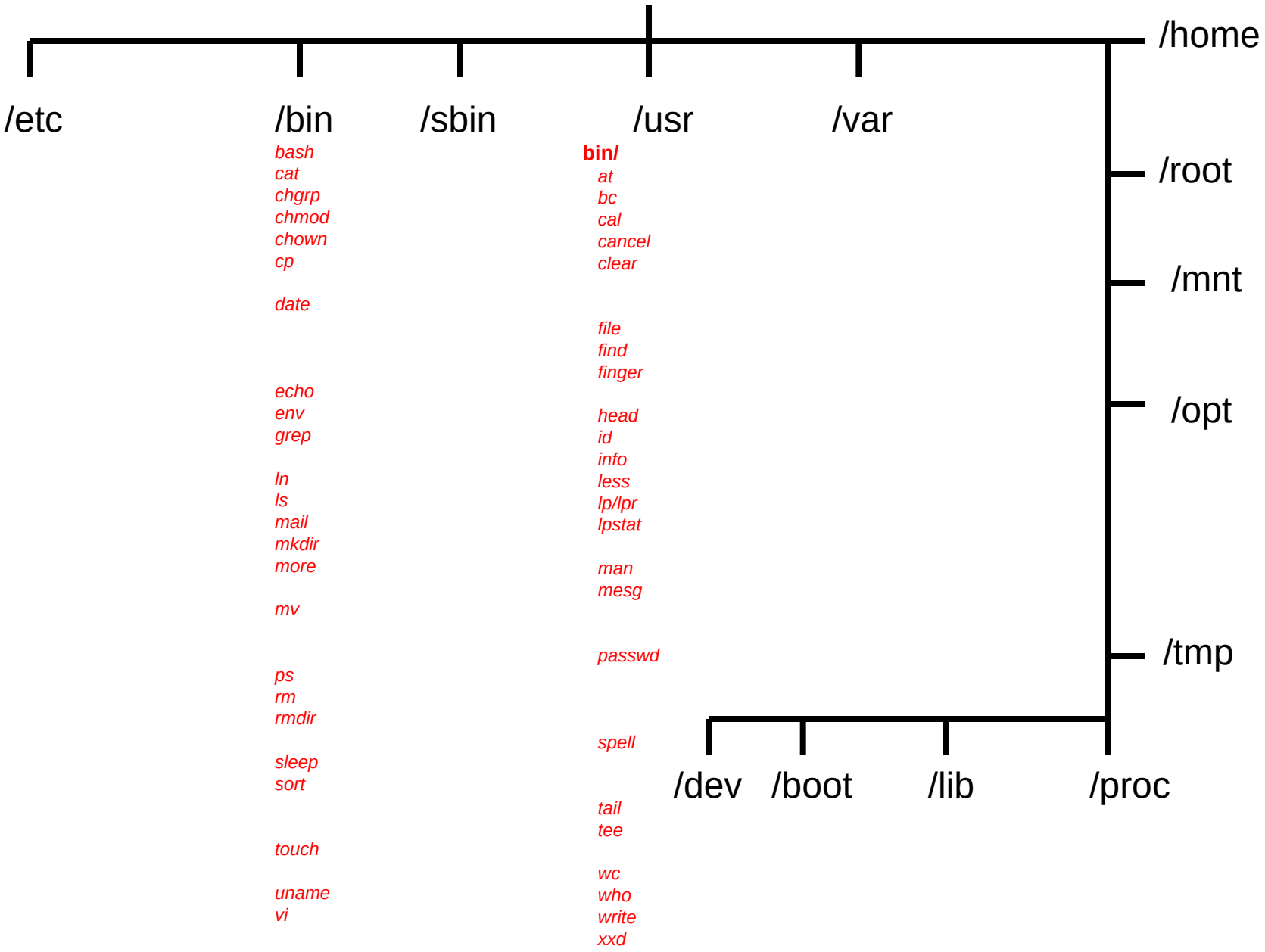
Example GNU/Linux Directory Structure




```
[root@tomcat ~]# ls /
bin   dev  home  lost+found  misc  net  proc  sbin     srv  tmp  var
boot  etc  lib   media       mnt   opt  root  selinux  sys  usr
[root@tomcat ~]#
```

Example GNU/Linux Directory Structure

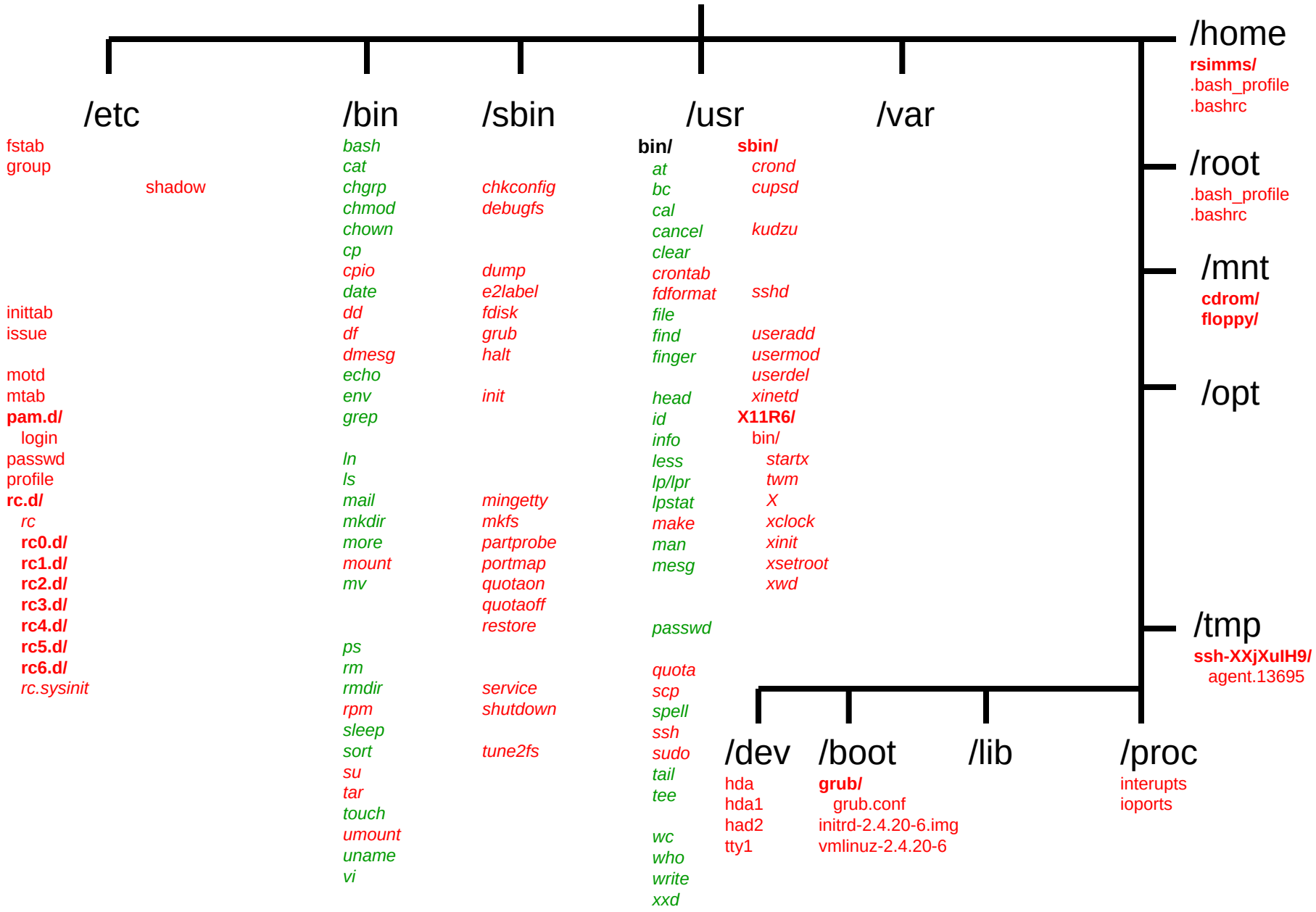
CIS 90 files, directories, commands



Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset

Example GNU/Linux Directory Structure

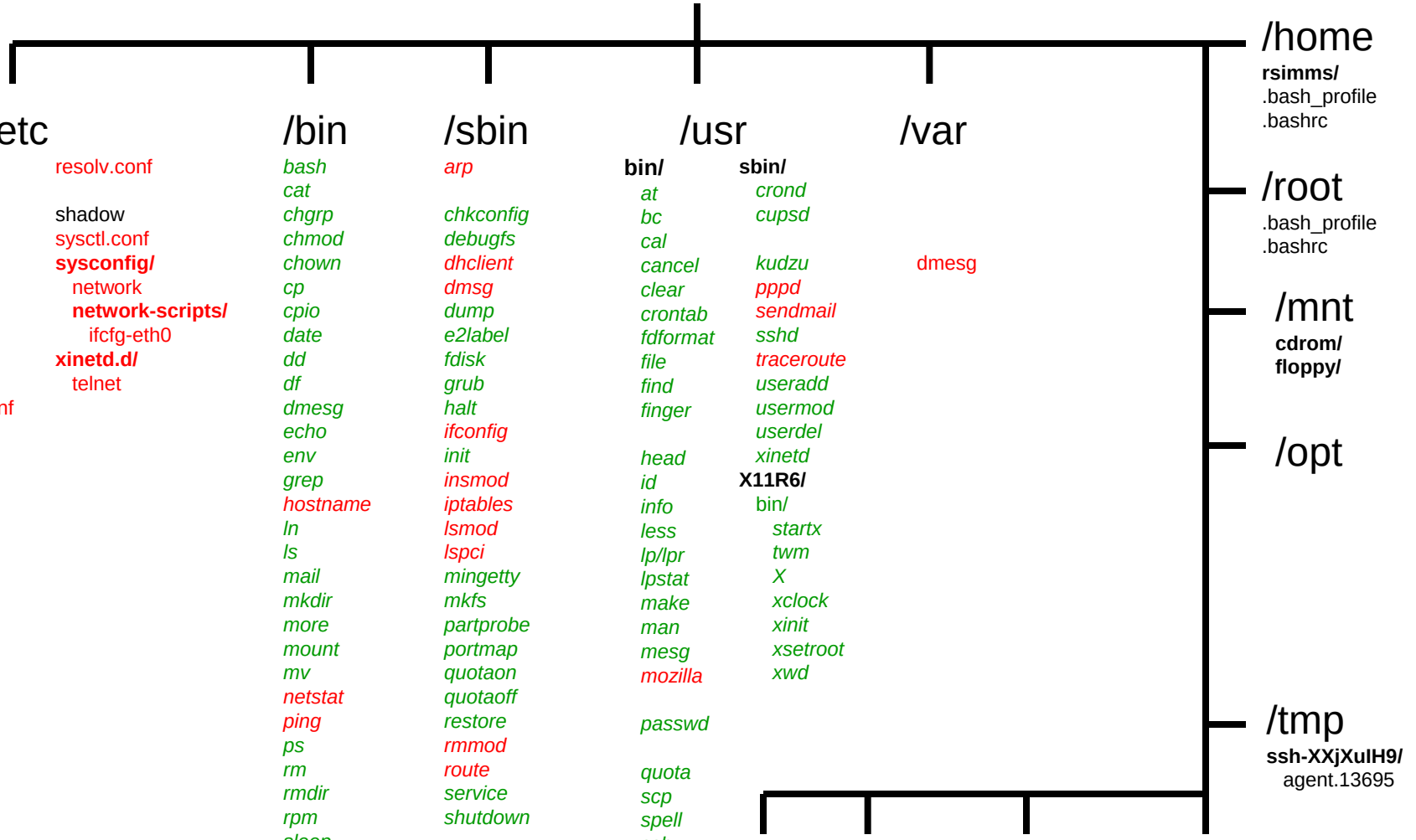
CIS 191 files, directories, commands



Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset

Example GNU/Linux Directory Structure

/



/etc

fstab
group
hosts
hosts.allow
hosts.deny
inittab
issue
modules.conf
motd
mtab
pam.d/
  login
passwd
profile
rc.d/
  rc
  rc0.d/
  rc1.d/
  rc2.d/
  rc3.d/
  rc4.d/
  rc5.d/
  rc6.d/
  rc.sysinit
resolv.conf
shadow
sysctl.conf
sysconfig/
  network
  network-scripts/
    ifcfg-eth0
xinetd.d/
  telnet

/bin

bash
cat
chgrp
chmod
chown
cp
cpio
date
dd
df
dmesg
echo
env
grep
hostname
ln
ls
mail
mkdir
more
mount
mv
netstat
ping
ps
rm
rmdir
rpm
sleep
sort
su
tar
touch
umount
uname
vi

/sbin

arp
chkconfig
debugfs
dhclient
dmsg
dump
e2label
fdisk
grub
halt
ifconfig
init
insmod
iptables
lsmod
lspci
mingetty
mkfs
partprobe
portmap
quotaon
quotaoff
restore
rmmod
route
service
shutdown
tune2fs

/usr

bin/
  at
  bc
  cal
  cancel
  clear
  crontab
  fdformat
  file
  find
  finger
  head
  id
  info
  less
  lp/lpr
  lpstat
  make
  man
  mesg
  mozilla
  passwd
  quota
  scp
  spell
  ssh
  sudo
  tail
  tee
  telnet
  wc
  who
  write
  xxd

sbin/
  crond
  cupsd
  kudzu
  pppd
  sendmail
  sshd
  traceroute
  useradd
  usermod
  userdel
  xinetd

X11R6/
  bin/
    startx
    twm
    X
    xclock
    xinit
    xsetroot
    xwd

/var

dmesg

/home

rsimms/
.bash_profile
.bashrc

/root

.bash_profile
.bashrc

/mnt

cdrom/
floppy/

/opt

/tmp

ssh-XXjXuIH9/
  agent.13695

/dev

hda
hda1
had2
tty1

/boot

grub/
  grub.conf
initrd-2.4.20-6.img
vmlinuz-2.4.20-6

/lib

modules/
  2.4.20-6/
    kernel/
      drivers/
        net/
          3c59x.o

/proc

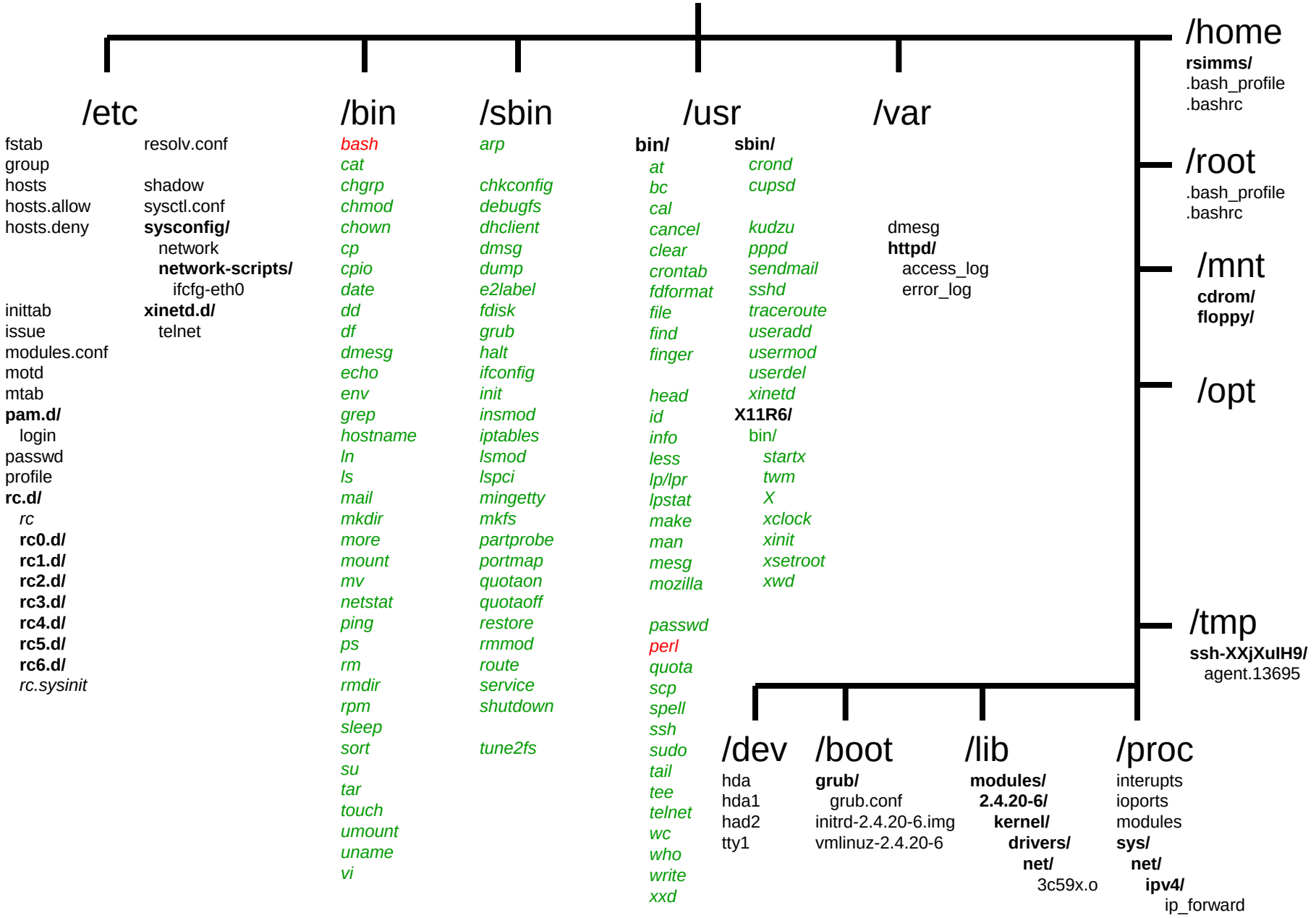
interupts
ioports
modules
sys/
  net/
    ipv4/
      ip_forward

Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset

Example GNU/Linux Directory Structure

/

CIS 130 files, directories, commands

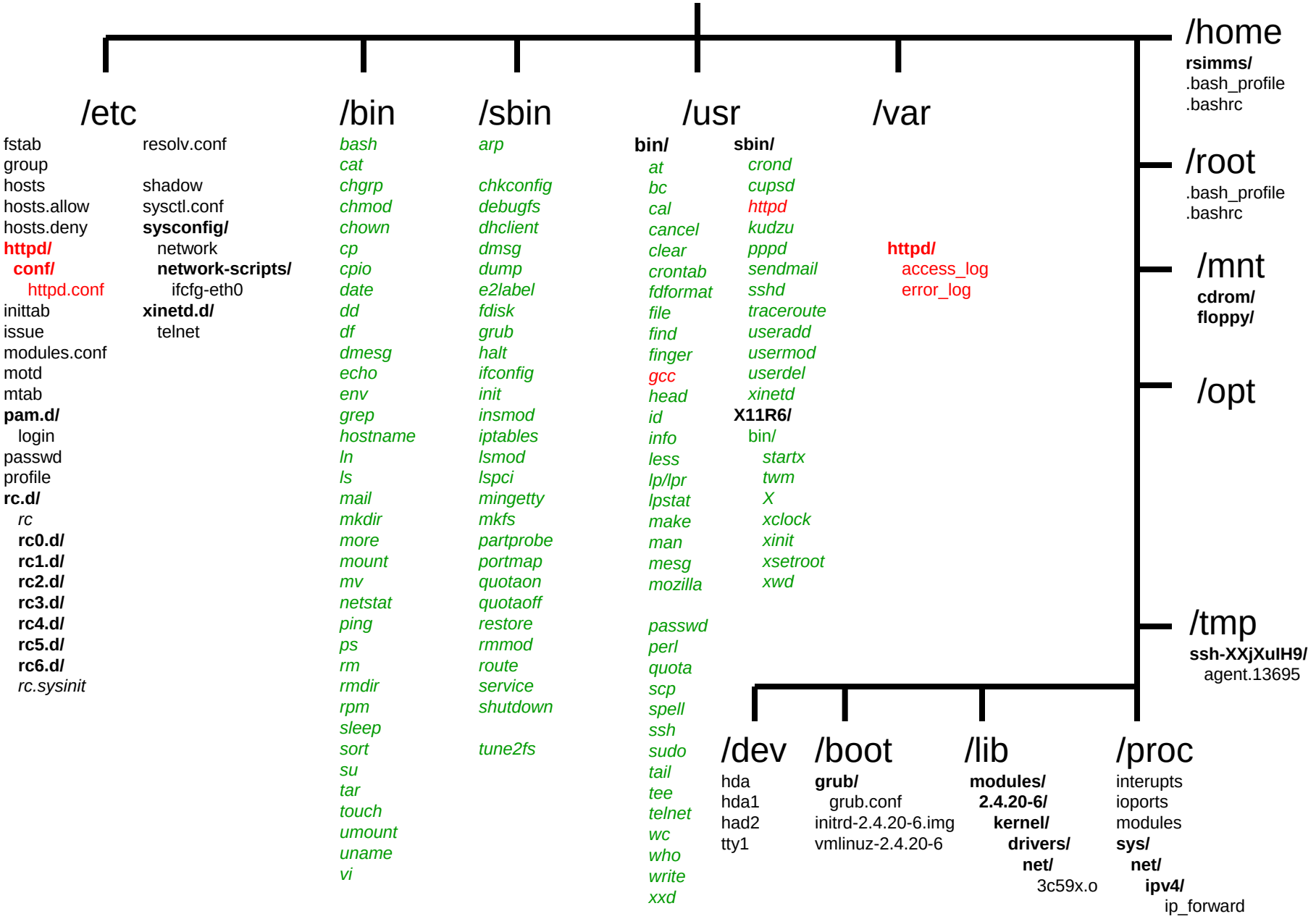


Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset shell keywords = if, then, else, case, for, while

Example GNU/Linux Directory Structure

/

CIS 164 files, directories, commands



Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset shell keywords = if, then, else, case, for, while

Example GNU/Linux Directory Structure

(showing just a few of the many files)

/

CIS 165PH files, directories, commands

/etc

fstab
group
hosts
hosts.allow
hosts.deny
httpd/
conf/
httpd.conf
inittab
issue
modules.conf
motd
mtab
pam.d/
login
passwd
profile
rc.d/
rc
rc0.d/
rc1.d/
rc2.d/
rc3.d/
rc4.d/
rc5.d/
rc6.d/
rc.sysinit
resolv.conf
shadow
sysctl.conf
sysconfig/
network
network-scripts/
ifcfg-eth0
xinetd.d/
telnet

/bin

bash
cat
chgrp
chmod
chown
cp
cpio
date
dd
df
dmesg
echo
env
grep
hostname
ln
ls
mail
mkdir
more
mount
mv
netstat
ping
ps
rm
rmdir
rpm
sleep
sort
su
tar
touch
umount
uname
vi

/sbin

arp

chkconfig
debugfs
dhclient
dmsg
dump
e2label
fdisk
grub
halt
ifconfig
init
insmod
iptables
lsmod
lspci
mingetty
mkfs
partprobe
portmap
quotaon
quotaoff
restore
rmmod
route
service
shutdown

tune2fs

/usr

bin/
at
bc
cal
cancel
clear
crontab
fdformat
file
find
finger
gcc
head
id
info
less
lp/lpr
lpstat
make
man
mesg
mozilla

passwd
perl
quota
scp
spell
ssh
sudo
tail
tee
telnet
wc
who
write
xxd

sbin/
crond
cupsd
httpd
kudzu
pppd
sendmail
sshd
traceroute
useradd
usermod
userdel
xinetd
X11R6/
bin/
startx
twm
X
xclock
xinit
xsetroot
xwd

/var

httpd/
access_log
error_log

/home

rsimms/
.bash_profile
.bashrc

/root

.bash_profile
.bashrc

/mnt

cdrom/
floppy/

/opt

lampp/
bin
mysql
htdocs
phpmyadmin/
index.php
sbin/
mysqld

/tmp

ssh-XXjXuIH9/
agent.13695

/dev

hda
hda1
had2
tty1

/boot

grub/
grub.conf
initrd-2.4.20-6.img
vmlinuz-2.4.20-6

/lib

modules/
2.4.20-6/
kernel/
drivers/
net/
3c59x.o

/proc

interupts
ioports
modules
sys/
net/
ipv4/
ip_forward

Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset shell keywords = if, then, else, case, for, while

Example GNU/Linux Directory Structure

(showing just a few of the many files)

/

CIS 193 files, directories, commands

/etc

fstab
group
hosts
hosts.allow
hosts.deny
httpd/
conf/
httpd.conf
inittab
issue
modules.conf
motd
mtab
pam.d/
login
passwd
profile
rc.d/
rc
rc0.d/
rc1.d/
rc2.d/
rc3.d/
rc4.d/
rc5.d/
rc6.d/
rc.sysinit
resolv.conf
securetty
shadow
sysctl.conf
sysconfig/
network
network-scripts/
ifcfg-eth0
xinetd.d/
telnet

/bin

bash
cat
chgrp
chmod
chown
cp
cpio
date
dd
df
dmesg
echo
env
grep
hostname
ln
ls
mail
mkdir
more
mount
mv
netstat
ping
ps
rm
rmdir
rpm
sleep
sort
su
tar
touch
umount
uname
vi

/sbin

arp
bastille
chkconfig
debugfs
dhclient
dmsg
dump
e2label
fdisk
grub
halt
ifconfig
init
insmod
iptables
lsmod
lspci
mingetty
mkfs
partprobe
portmap
quotaon
quotaoff
restore
rmmod
route
service
shutdown
tripwire
tune2fs

/usr

bin/
at
bc
cal
cancel
clear
crontab
fdformat
file
find
finger
gcc
head
id
info
less
lp/lpr
lpstat
make
man
mesg
mozilla
openssl
passwd
perl
quota
scp
spell
ssh
sudo
tail
tee
telnet
wc
who
write
xxd

sbin/
crond
cupsd
httpd
kudzu
pppd
sendmail
sshd
traceroute
useradd
usermod
userdel
xinetd

X11R6/
bin/
startx
twm
X
xclock
xinit
xsetroot
xwd

/var

log/
Bastille/
Assessment/
assessment-report.html
dmesg
httpd/
access_log
error_log
spool/
clientmqueue

/home

rsimms/
.bash_profile
.bashrc

/root

.bash_profile
.bashrc

/mnt

cdrom/
floppy/

/opt

lampp/
bin
mysql
htdocs
phpmyadmin/
index.php
sbin/
mysqld

/tmp

ssh-XXjXulH9/
agent.13695

/dev

hda
hda1
had2
tty1

/boot

grub/
grub.conf
initrd-2.4.20-6.img
vmlinuz-2.4.20-6

/lib

modules/
2.4.20-6/
kernel/
drivers/
net/
3c59x.o

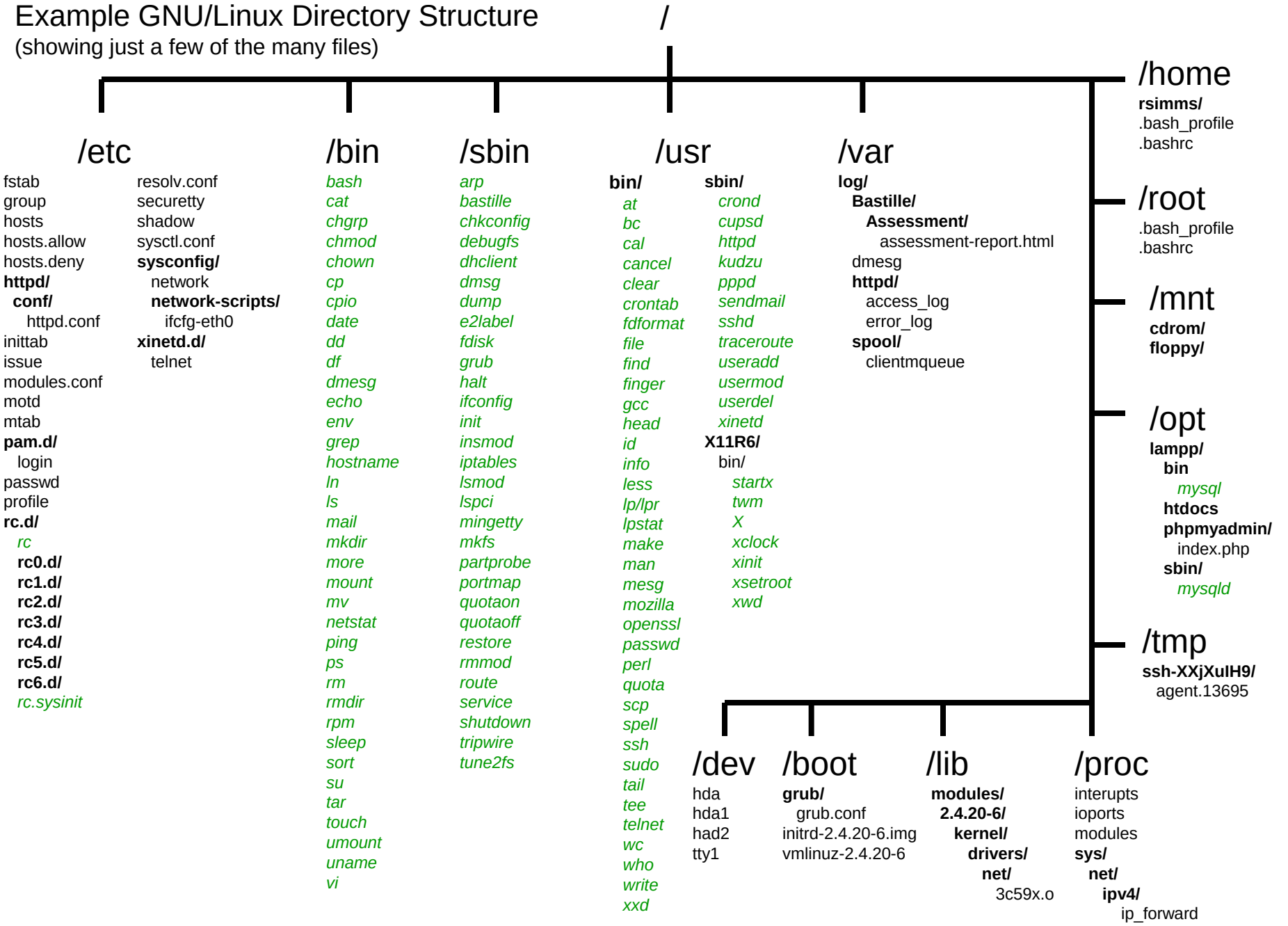
/proc

interupts
ioports
modules
sys/
net/
ipv4/
ip_forward

Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset shell keywords = if, then, else, case, for, while

Example GNU/Linux Directory Structure

(showing just a few of the many files)



Note: shell builtins = cd, echo, exit, export, history, jobs, kill, pwd, set, type, umask, unset shell keywords = if, then, else, case, for, while

Class Field Trip

1) boot
   - The kernel

2) etc
   - Apache web configuration file
   - motd
   - passwd

3) var
   - mail
   - www

4) home
   - Student home directories
   - depot
   - bin

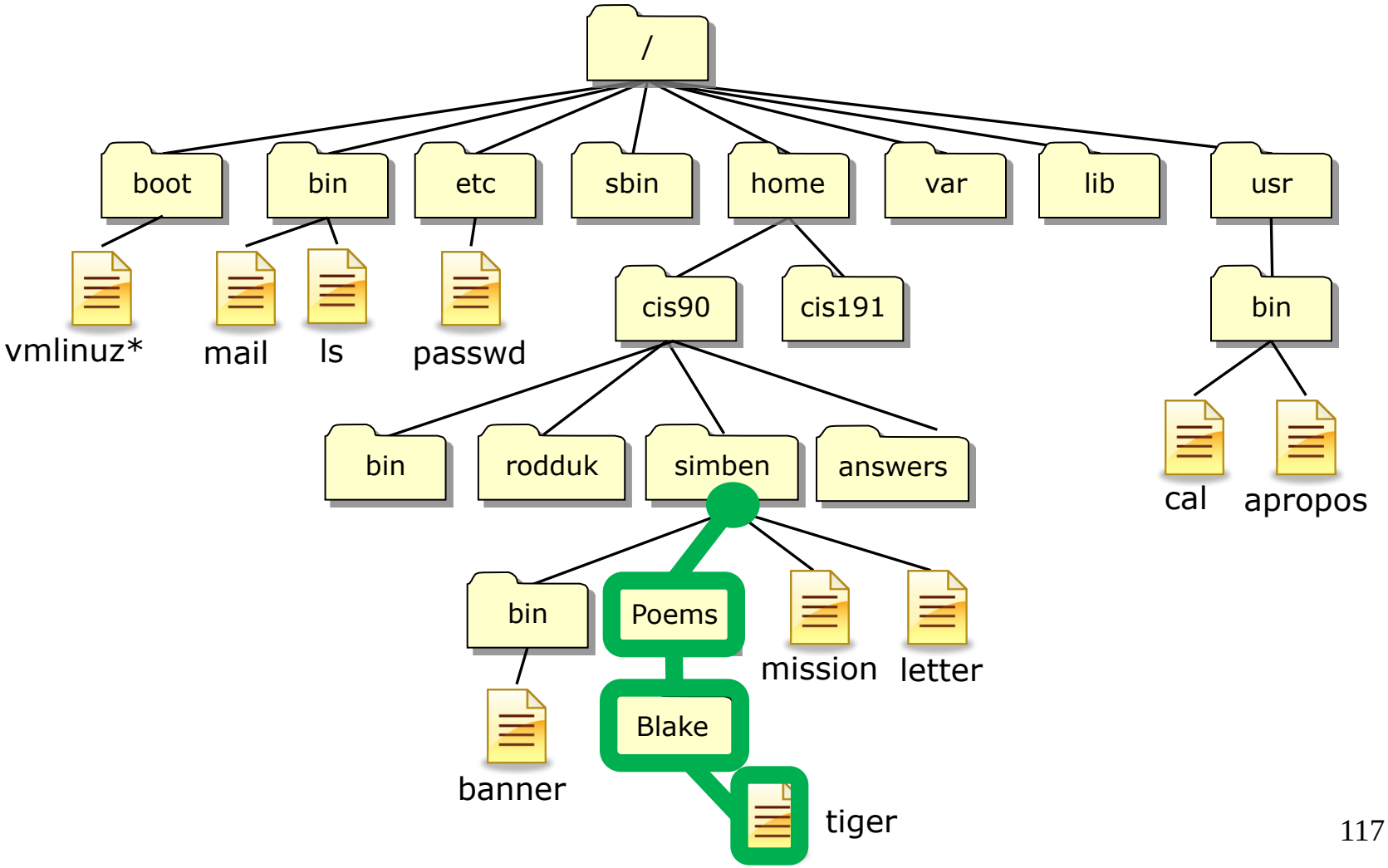
Navigating the UNIX file tree

Navigating the tree

- Use the **cd** command to change directories *(your legs)*
- Use the **ls** command to list files at your current location (your eyes)
- Use the **pwd** command to check where you are (your GPS)

*Note, as CIS 90 students your command prompt has been configured to show what you would normally get with the **pwd** command. As you move around the tree your command prompt will change to show your current location.*

How do we walk the tree from our home directory to the directory containing the tiger file and print it?



Class Exercise

```
/home/cis90/simben $ cd
/home/cis90/simben $ ls
bigfile      Hidden         log            proposal1   text.err
bin          lab01.graded   mbox           proposal2   text.fxd
countargs    Lab2.0         Miscellaneous  proposal3   timecal
dead.letter  Lab2.1         mission        small_town  uhistory
empty        letter         Poems          spellk      what_am_i

/home/cis90/simben $ cd Poems/
/home/cis90/simben/Poems $ ls
ant  Blake  nursery  Shakespeare  twister  Yeats

/home/cis90/simben/Poems $ cd Blake/
/home/cis90/simben/Poems/Blake $ ls
jerusalem  tiger
/home/cis90/simben/Poems/Blake $ cat tiger
Tiger, Tiger burning bright
In the forest of the night,
What immortal hand or eye
Dare frame thy fearful symmetry?
```

start in our home directory

see what's there

go down into Poems directory

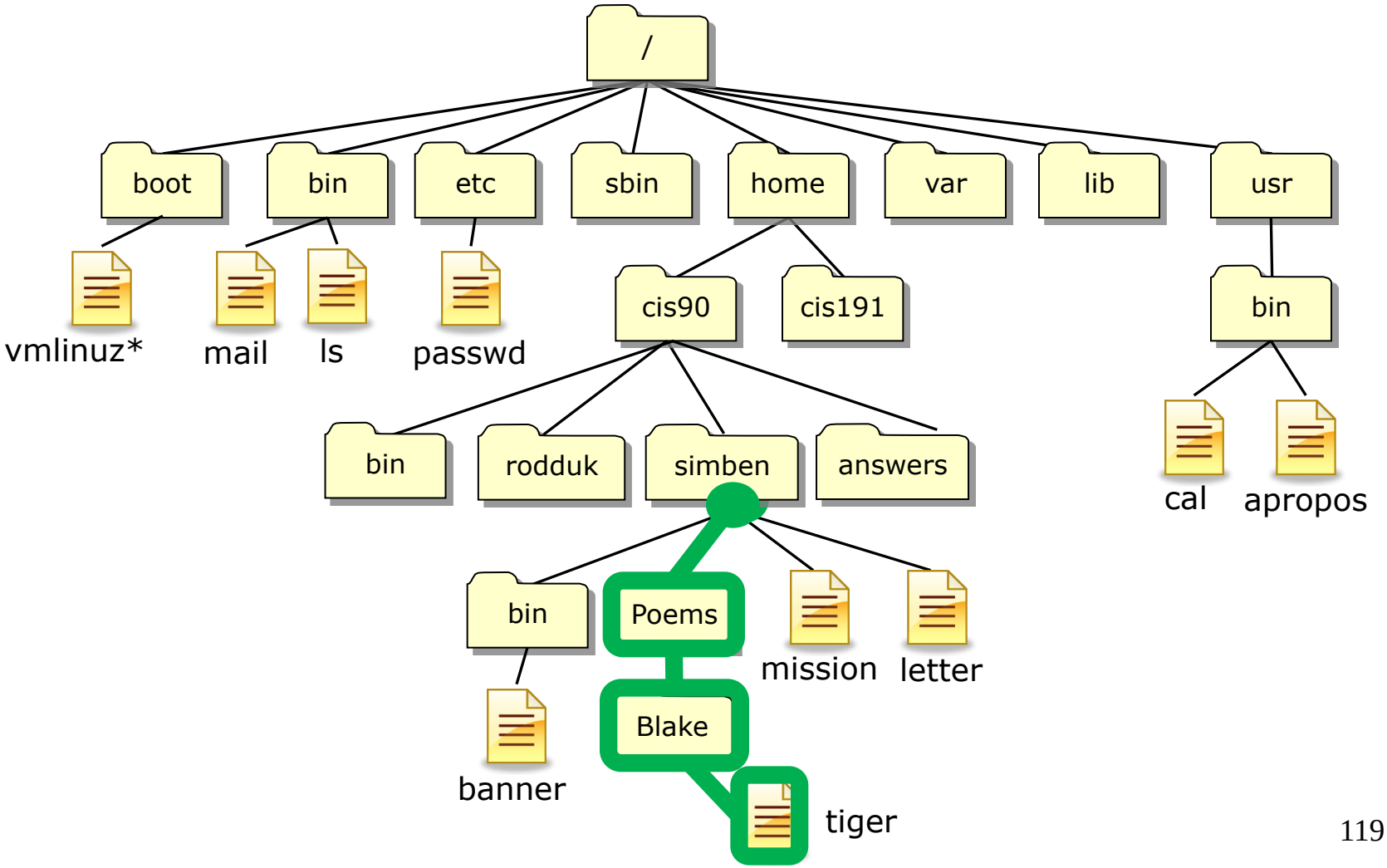
see what's there

go down into Blake directory

see what's there

print tiger file

Alternatively how could we print the tiger file from our home directory without navigating there first?



Class Exercise

```
/home/cis90/simben $ cd
/home/cis90/simben $ cat Poems/Blake/tiger
Tiger, Tiger burning bright
In the forest of the night,
What immortal hand or eye
Dare frame thy fearful symmetry?
/home/cis90/simben $
```

start in our home directory

using a relative pathname

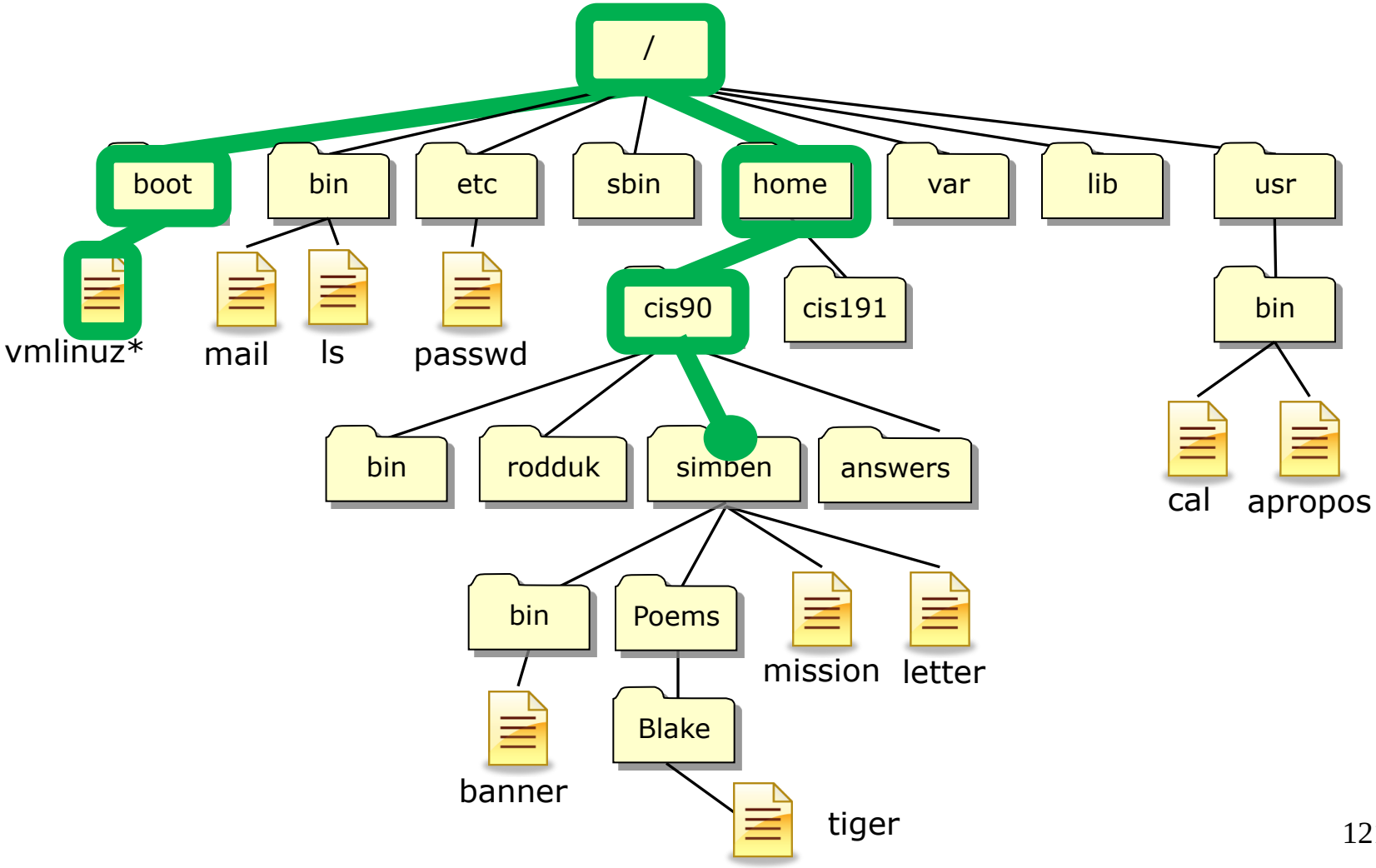
```
/home/cis90/simben $ cat /home/cis90/simben/Poems/Blake/tiger
Tiger, Tiger burning bright
In the forest of the night,
What immortal hand or eye
Dare frame thy fearful symmetry?
/home/cis90/simben $
```

using an absolute pathname

```
/home/cis90/simben $ cat tiger
cat: tiger: No such file or directory
/home/cis90/simben $
```

using an incorrect pathname (the tiger file is not in our home directory)

How do we walk the tree from our home directory to the directory containing the Linux kernel and do a long listing of it?



Class Exercise

```
/home/cis90/simben/Poems/Blake $ cd
```
start in your home directory

```
/home/cis90/simben $ cd ..
```
go up the tree

```
/home/cis90 $ ls
```
look around

```
ahrmat   bodian  colabd  flamat  hovdav  macrya  milmic  phacha  pummas  shidev
answers  bunsol  deltas  gueous  huljef  maxsco  olscam  plajos  rafdav  simben
bin      cheken  depot   guest   jimmel  mcidar  otteve  plajua  reedie  varana
blerav   cofcol  doucor  helrog  lowmic  milhen  pacnan  porjon  rodduk  veleli
```
student home directories

```
/home/cis90 $ cd ..
```
go up

```
/home $ ls
```
look around

our class directory — *my home directory* — *where labs are submitted*

```
backup  cis172  cis192  cis90    cis98     guest  mikki  rsimms  turnin
cis164  cis191  cis193  cis90ol  gerlinde  jimg   rick   ryan
```

```
/home $ cd ..
```
go up

```
/ $ ls
```
look around

```
bin   dev  home  lost+found  misc  net  proc  sbin     srv  tftpboot  u    var
boot  etc  lib   media       mnt   opt  root  selinux  sys  tmp       usr
```

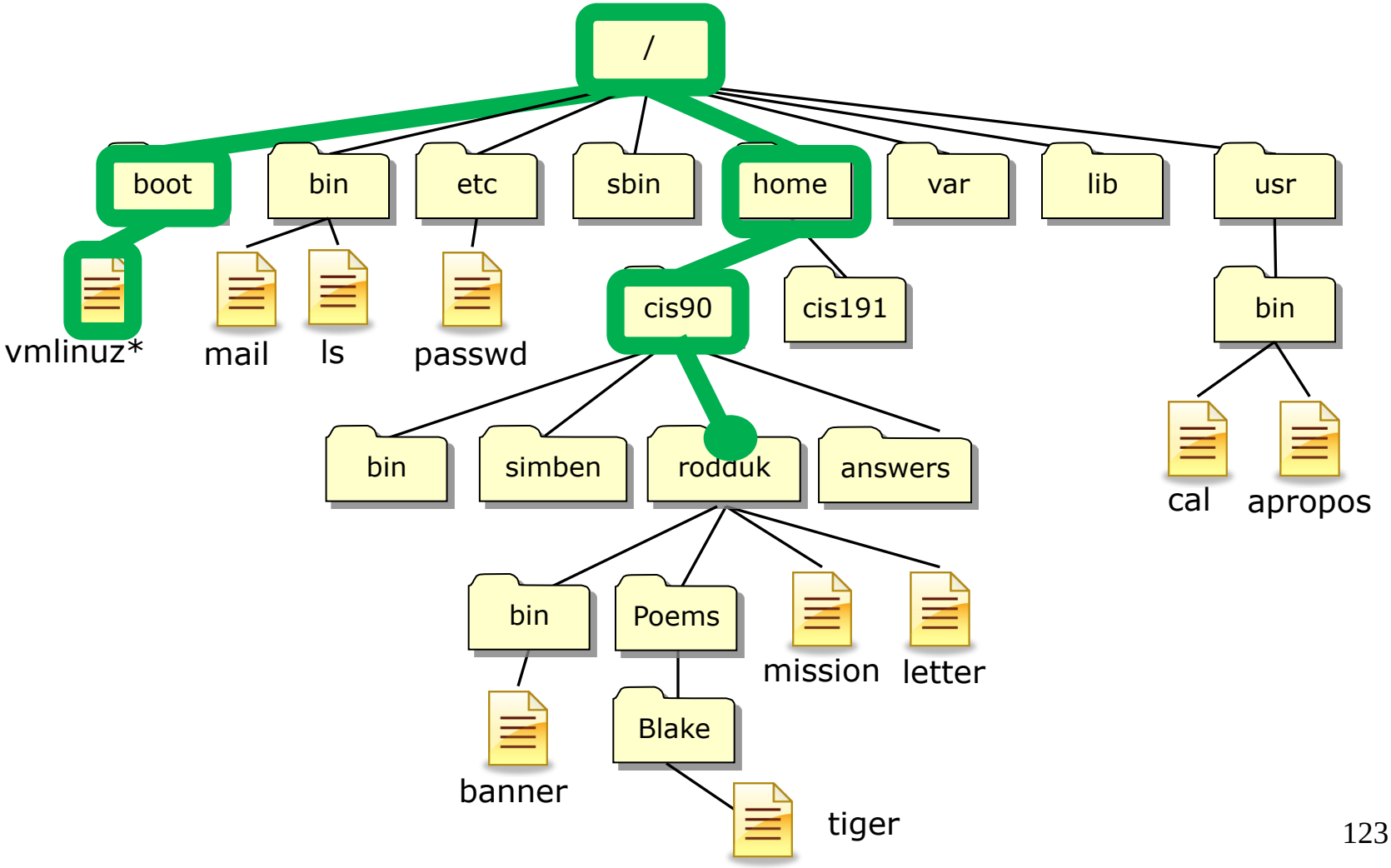
```
/ $ cd boot
```
go down into boot

```
/boot $ ls
```
look around

```
config-2.6.18-164.el5      lost+found                vmlinuz-2.6.18-164.el5
grub                       symvers-2.6.18-164.el5.gz
initrd-2.6.18-164.el5.img  System.map-2.6.18-164.el5
```
the Linux kernel

```
/boot $ ls -l vmlinuz-2.6.18-164.el5
-rw-r--r-- 1 root root 1855956 Aug 18  2009 vmlinuz-2.6.18-164.el5
```

Alternatively, how could we do the same thing without walking there first?



Class Exercise

```
/home/cis90/simben/Poems/Blake $ cd
/home/cis90/simben $ ls -l /boot/vmlinuz-2.6.18-164.el5
-rw-r--r-- 1 root root 1855956 Aug 18  2009 /boot/vmlinuz-2.6.18-164.el5
/home/cis90/simben $
```

start in your home directory

using an absolute pathname

the Linux kernel

Navigating

cd command (your legs)

cd command
change directory

- Syntax: **cd [*directory*]**
- Changes the current working directory to the directory specified.
- Use **cd** with no arguments to return to your home directory.

  Note, users always start in their home directory after logging in.
  Every user's home directory is configured in the /etc/passwd file.

- The *directory* can be:
  An absolute pathname, e.g. **cd /home/cis90/duke/Poems/ant**
  A relative pathname, e.g. **cd Poems, cd Poems/Yeats**
  A **..** for the parent of the current working directory, e.g. **cd ..**

- Note, **cd** is a Bash builtin command (part of the shell itself)
  /home/cis90/simben $ **type cd**
  cd is a shell builtin

More on .. and .

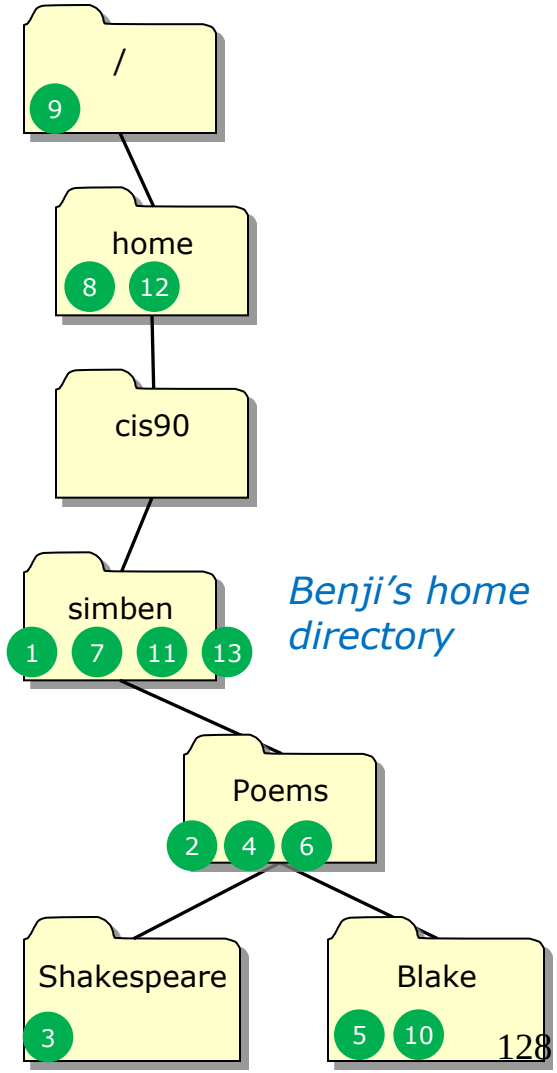
To move up the tree use: **cd ..**

.. is a hidden file located in every single directory and it is hard linked to the absolute pathname of the parent directory

cd command

change directory example

```
/home/cis90/simmen $ echo $HOME
/home/cis90/simben
/home/cis90/simmsben $ echo $PS1
$PWD $
1  /home/cis90/simmen $ cd Poems/
2  /home/cis90/simben/Poems $ cd Shakespeare/
3  /home/cis90/simben/Poems/Shakespeare $ cd ..
4  /home/cis90/simben/Poems $ cd Blake/
5  /home/cis90/simben/Poems/Blake $ cd ..
6  /home/cis90/simben/Poems $ cd ..
7  /home/cis90/simben $ cd /home
8  /home $ cd ..
9  / $ cd /home/cis90/simben/Poems/Blake/
10 /home/cis90/simben/Poems/Blake $ cd
11 /home/cis90/simben $ cd ../../
12 /home $ cd
13 /home/cis90/simben $
```



Navigating

pwd command (your GPS)

pwd command
print working directory

- The **pwd** command is your "GPS" to show your current location on the UNIX file tree. Especially with more typical prompts!

- The **pwd** command is equivalent to displaying the value of the PWD environment variable

```
[rsimms@opus net]$ pwd
/lib/modules/2.6.18-164.el5/kernel/drivers/net
```

This is a UNIX command

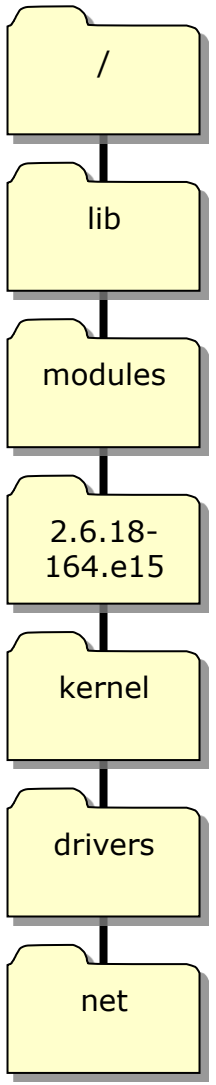
```
[rsimms@opus net]$ echo $PWD
/lib/modules/2.6.18-164.el5/kernel/drivers/net
```

This is a UNIX command

This is shell environment variable (used as an argument to the echo command)

Note: The default shell prompt CIS 90 students utilizes the PWD variable to always show the current working directory.

i.e. When CIS 90 students login this command: PS1= '$PWD $ ' is automatically done as part of setting up their shell environment.



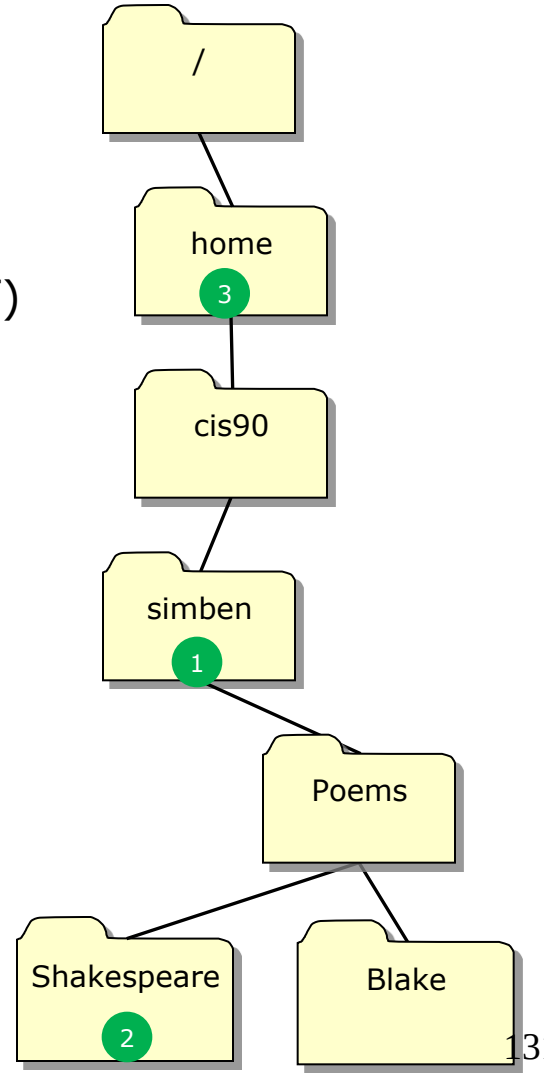
pwd command

print working directory

Note: The shell prompt has been configured for CIS 90 students to always show the current working directory. This example shows the pwd command with a more typical prompt.

- Syntax: **pwd**
- Prints the current working directory.
- pwd is a BASH builtin command (part of the shell itself)
  /home/cis90/simben $ **type pwd**
  pwd is a shell builtin

```
/home/cis90/simben $ PS1='[\u@\h \W]\$ '
1 [simben90@opus ~]$ pwd
/home/cis90/simben
[simben90@opus ~]$ cd Poems/Shakespeare/
2 [simben90@opus Shakespeare]$ pwd
/home/cis90/simben/Poems/Shakespeare
[simben90@opus Shakespeare]$ cd /home/
3 [simben90@opus home]$ pwd
/home
/home/cis90/simben $ PS1='$PWD $ '
/home/cis90/simben $
```



Navigating

ls command (your eyes)

Important commands for your toolbox:

- ls *list files*

ls command

- Syntax: **ls [options] [*directory*]...**

Option	Description
-a	Show all files, even the hidden ones with names starting with "."
-i	Show inode numbers
-d	Show the directory itself rather than the contents of the directory
-l	Long listing (lots of inode information)
-F	Show file types (directory/, program*, link@, socket=)
-S	Sort by size
-R	Recursive (show all sub-directories)

- The *directory* argument can be:
  An absolute pathname, e.g. **cd /home/cis90/duke/Poems/**
  A relative pathname, e.g. **cd /Poems**
  If no directory is specified, the current working directory is used.
  More than one directory can be specified

- Use **man ls** to see more information.

ls command

List Files

FYI …

- **ls** is in /bin and has been aliased to use color on terminal output

```
[simmsben@opus ~]$ type -a ls
ls is aliased to `ls --color=tty'
ls is /bin/ls
```

Using the type command to show where a command resides on the path

Note: the `--color=tty` is an option on the **ls** command. Options that are fully spelled usually use two dashes -- instead of 1

We will learn about aliases later in the course (unless your read Melissa's post on Lab 3!)

ls command example

with no options

Regular files in black

```
/home/cis90/simmsben $ ls
bigfile  Hidden  letter         Poems      proposal3   text.err  what_am_i
bin      Lab2.0  Miscellaneous  proposal1  small_town  text.fxd
empty    Lab2.1  mission        proposal2  spellk      timecal
```

Directories in blue

Executables (programs or scripts) in green

*Using the **ls** command with no arguments will list the files in the current directory*

ls command example

with the -F option

Regular files have no suffix

```
/home/cis90/simmsben $ ls -F
bigfile   Hidden/   letter           Poems/      proposal3   text.err   what_am_i
bin/      Lab2.0/   Miscellaneous/   proposal1   small_town  text.fxd
empty     Lab2.1/   mission          proposal2   spellk      timecal*
```

Directories end with /

*Executables (programs or scripts) end with **

*Use the **–F** option to show file types with symbols rather than color (helpful if you are color blind)*

ls command example

with the -a option

```
/home/cis90/simmsben $ cd
```

***cd** with no arguments takes you to your home directory*

```
/home/cis90/simmsben $ ls -a
.               .bashrc        Hidden     Miscellaneous  proposal1   text.err
..              bigfile        Lab2.0     mission        proposal2   text.fxd
.bash_history   bin            Lab2.1     .mozilla       proposal3   timecal
.bash_logout    .emacs         .lesshst   .plan          small_town  what_am_i
.bash_profile   empty          letter     Poems          spellk      .zshrc
/home/cis90/simmsben $
```

Use the –a option to show hidden files (files whose names start with a ".")

***..** is the parent directory*

***.** is this the current directory, think of . as meaning "here"*

ls command example

with the -S option

```
/home/cis90/simben $ ls -lS
total 344
-rw------- 1 simben90 cis90 113773 Feb 28 22:35 mbox
-rw-rw-r-- 1 simben90 cis90  21295 Feb 28 20:46 uhistory
-rw-r--r-- 2 simben90 cis90  10576 Jul 20  2001 bigfile
drwxr-xr-x 2 simben90 cis90   4096 Feb 12 16:07 bin
d--------- 2 simben90 cis90   4096 Feb  1  2002 Hidden
drwxr-xr-x 2 simben90 cis90   4096 Feb 17  2001 Lab2.0
drwxr-xr-x 3 simben90 cis90   4096 Feb 17  2001 Lab2.1
drwxr-xr-x 2 simben90 cis90   4096 Sep 11  2005 Miscellaneous
drwxr-xr-x 5 simben90 cis90   4096 Jan 18  2004 Poems
-rw-r--r-- 1 simben90 cis90   2175 Jul 20  2001 proposal2
-rw-r--r-- 1 simben90 cis90   2054 Sep 14  2003 proposal3
-r-------- 1 simben90 staff   1873 Feb 23 11:58 lab02.graded
-rw-r--r-- 1 simben90 cis90   1580 Nov 16  2004 small_town
-r-------- 1 simben90 staff   1182 Feb 16 13:17 lab01.graded
-rw-r--r-- 1 simben90 cis90   1074 Aug 26  2003 proposal1
-rw-r--r-- 1 simben90 cis90   1044 Jul 20  2001 letter
-rw-r--r-- 1 simben90 cis90    759 Jun  6  2002 mission
-rw-r--r-- 1 simben90 cis90    572 Feb 22 16:07 log
-rwxr-xr-x 1 simben90 cis90    509 Jun  6  2002 timecal
-rw-r--r-- 1 simben90 cis90    494 Feb 12 16:39 lab01-submitted
-rw-r--r-- 1 simben90 cis90    485 Aug 26  2003 spellk
-rw------- 1 simben90 cis90    355 Feb 24 15:40 dead.letter
-rw-r--r-- 1 simben90 cis90    352 Jul 20  2001 what_am_i
-rw-r--r-- 1 simben90 cis90    250 Jul 20  2001 text.err
-rw-r--r-- 1 simben90 cis90    231 Jul 20  2001 text.fxd
-rw-r--r-- 1 simben90 cis90      0 Jul 20  2001 empty
/home/cis90/simben $
```

*Use the **-S** option to sort files by size*

ls command example

with the -i option

```
/home/cis90/simmsben $ cd
```

***cd** with no arguments take you to your home directory*

```
/home/cis90/simmsben $ ls -i
105056 bigfile   102566 Lab2.1          102608 proposal1   102613 text.err
102542 bin       102576 letter          102609 proposal2   102614 text.fxd
102551 empty     102577 Miscellaneous   102610 proposal3   102615 timecal
102552 Hidden    102582 mission         102611 small_town  102616 what_am_i
102555 Lab2.0    102584 Poems           102612 spellk
```

*Use the **-i** option to show the inode associated with a filename*

Question:

What are some different ways to get the inode number of your home directory?

Answer

```
/home/cis90/simben $ ls -id /home/cis90/simben/  using an absolute pathname
 98306 /home/cis90/simben/

/home/cis90/simben $ ls -i /home/cis90/  using contents of the parent directory
  98315 ahrmat    102261 depot     98572 macrya    98491 plajua
 102260 answers    98387 doucor    98435 maxsco    98499 porjon
2425560 bin        98395 flamat    98451 mcidar    98507 pummas
  98323 blerav    103280 gueous    98564 milhen    98515 rafdav
  98331 bodian    102852 guest     98459 milmic    98523 reedie
  98339 bunsol     98403 helrog    98637 olscam   131253 rodduk
  98355 cheken     98411 hovdav    98540 otteve    98588 shidev
  98363 cofcol     98419 huljef    98475 pacnan    98306 simben
  98645 colabd     98427 jimmel   102962 phacha    98596 varana
  98548 deltas     98556 lowmic    98483 plajos    98628 veleli

/home/cis90/simben $ ls -id .   The . is an absolute pathname to current directory
 98306 .
```

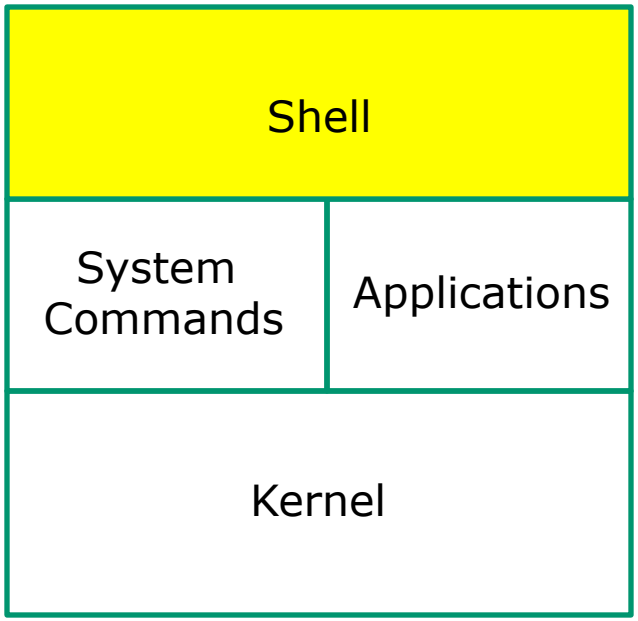
Class Exercise

- What is the name of your home directory?
- What is the inode number of your home directory?
- What file type is your home directory?
- What is the absolute path of your home directory?

*

metacharacter

Life of the Shell



1) Prompt
2) **Parse**
3) Search
4) Execute
5) Nap
6) Repeat

*Metacharacters, like the *, are processed and expanded during the Parse step*

(before the selected command is even run)

*

filename expansion metacharacter

- The * is a shell metacharacter
- During the parse steps the shell expands * and replaces it with matching filenames in the current directory or as part of any pathnames specified as arguments.
- The commands loaded by the shell never see the *, instead then see the expanded filenames.
- The * will match non-hidden filenames when used by itself.

*

filename expansion metacharacter

```
/home/cis90/simben/Poems/Yeats $ ls
mooncat  old  whitebirds
```

```
/home/cis90/simben/Poems/Yeats $ file mooncat old whitebirds
mooncat:    ASCII English text
old:        ASCII English text
whitebirds: ASCII English text
```

user manually types in each filename in directory

```
/home/cis90/simben/Poems/Yeats $ file *
mooncat:    ASCII English text
old:        ASCII English text
whitebirds: ASCII English text
```

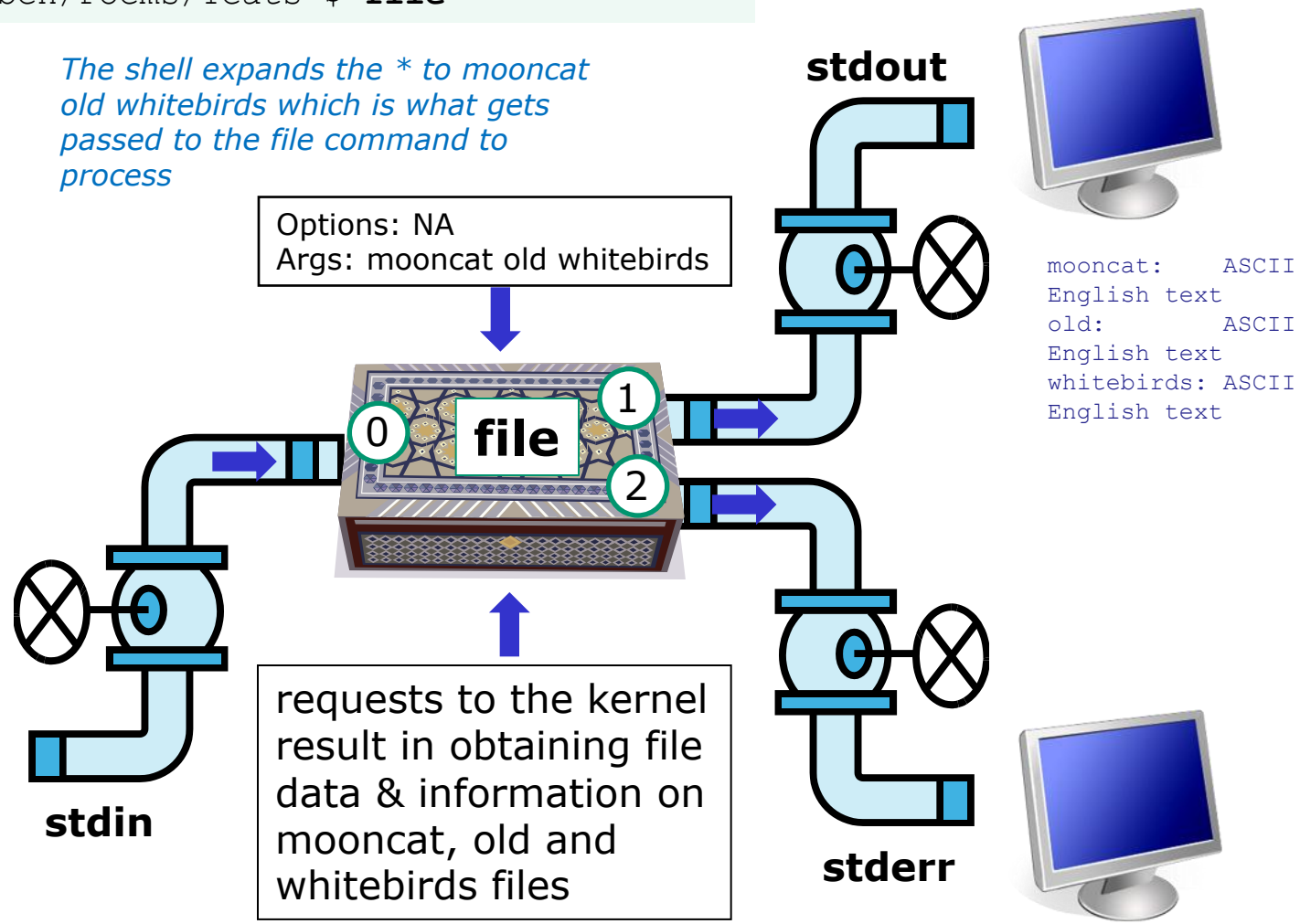
User lets the shell do the work instead

*In the second example, the shell, during the parse step, expands the * and replaces it with mooncat old whitebirds. The file command never sees the "*"*

Example program to process: file command

```
/home/cis90/simben/Poems/Yeats $ file *
```

*The shell expands the * to mooncat old whitebirds which is what gets passed to the file command to process*



* metacharacter

used as a prefix character

```
/home/cis90/simben $ ls
bigfile   Lab2.0          mission    proposal3   text.fxd
bin       Lab2.1          Poems      small_town  timecal
empty     letter          proposal1  spellk      what_am_i
Hidden    Miscellaneous   proposal2  text.err

/home/cis90/simben $ ls *.err
text.err
```

***.err** matches all file names **ending** with ".err"*

*Shell operation question: Does the **ls** command see the "*" typed by the user?*

* metacharacter

used as an infix character

```
/home/cis90/simben $ ls
bigfile   Lab2.0          mission    proposal3   text.fxd
bin       Lab2.1          Poems      small_town  timecal
empty     letter          proposal1  spellk      what_am_i
Hidden    Miscellaneous   proposal2  text.err

/home/cis90/simben $ ls *am*
what_am_i
```

am *matches all file names* ***containing*** *"am"*

Answer to the question on pervious slide: NO! The shell replaced the ".err" with the string "text.err" and that's what the* ***ls*** *command received as an argument.*

* metacharacter

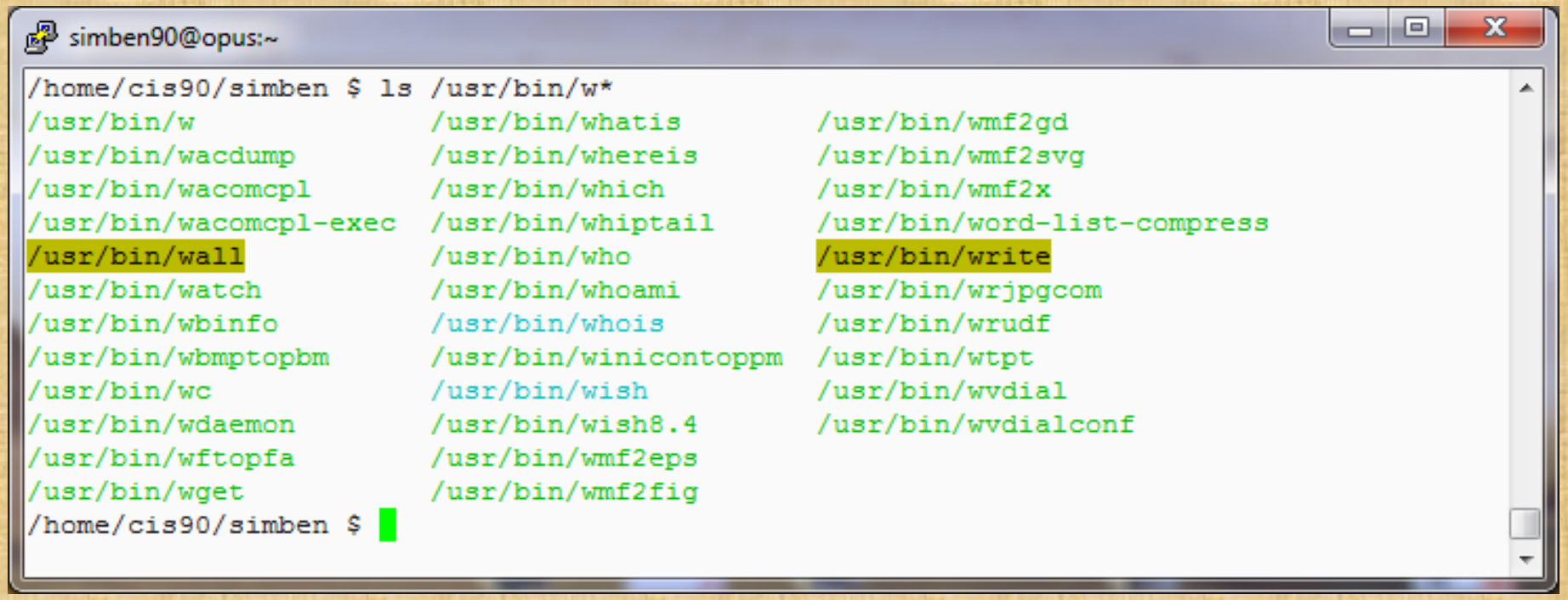
used as a postfix character

```
/home/cis90/simben $ ls
bigfile  Lab2.0          mission    proposal3   text.fxd
bin      Lab2.1          Poems      small_town  timecal
empty    letter          proposal1  spellk      what_am_i
Hidden   Miscellaneous   proposal2  text.err

/home/cis90/simmen $ ls p*
proposal1  proposal2  proposal3
```

p* *matches all file names* ***starting*** *with a "p"*

Class Activity

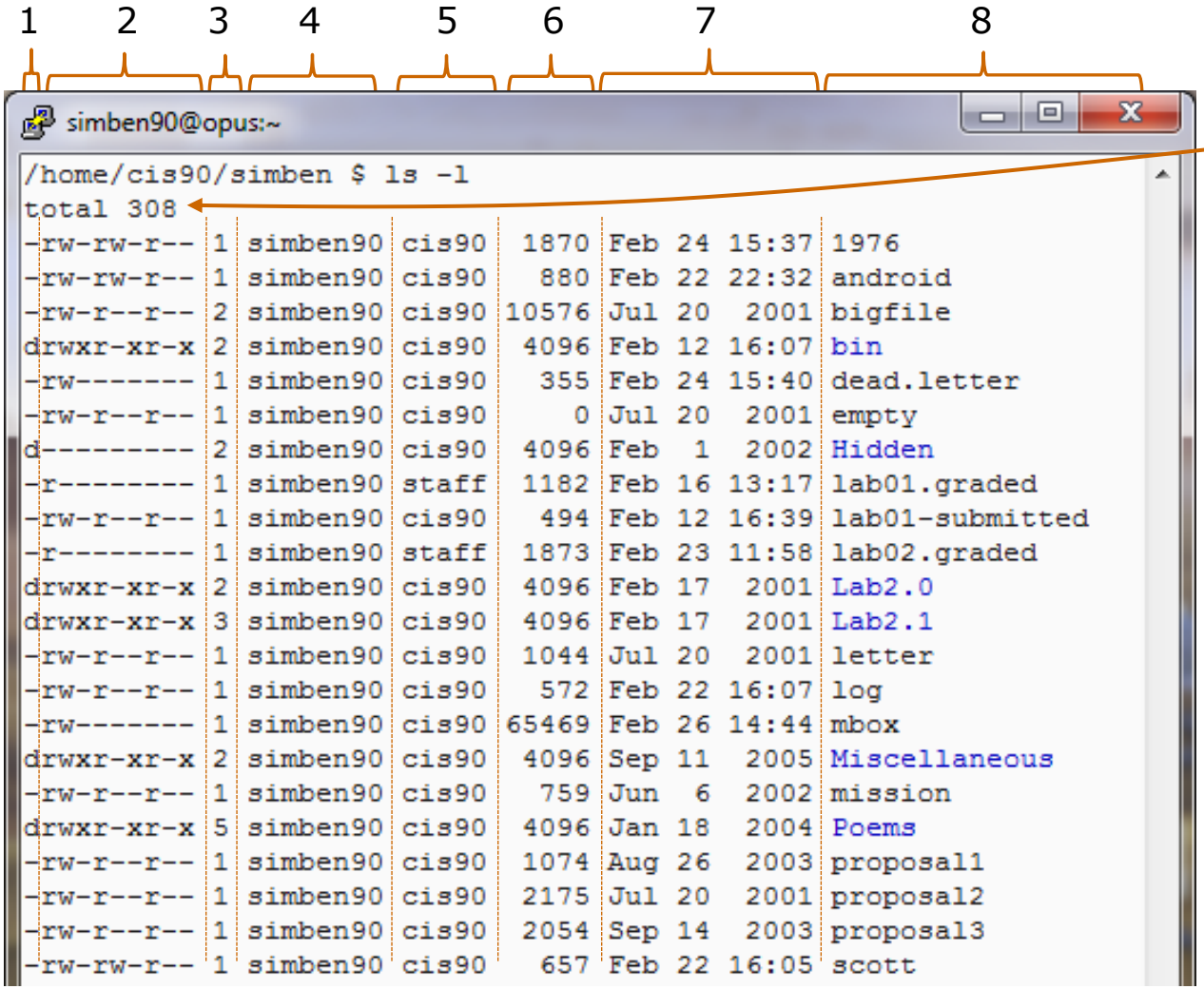
What commands in the /usr/bin directory starts with the letter w?



More on the ls command

ls command

Use the -l option for a "long listing"



total size of all files in blocks

On Opus, 1 block = 1024 bytes

1. file type
   - = regular
   d = directory
   l = link
2. permissions
3. number of hard links
4. owner
5. group
6. size (in bytes)
7. last modified
8. file name

ls command

Using files vs directories as arguments

With no arguments specified, all files in current directory will be listed

```
/home/cis90/simben $ ls
bigfile   Lab2.0          mission    proposal3   text.fxd
bin       Lab2.1          Poems      small_town  timecal
empty     letter          proposal1  spellk      what_am_i
Hidden    Miscellaneous   proposal2  text.err
```

```
/home/cis90/simben $ ls bigfile
bigfile
```

*With a **filename** specified as an argument, just that file will be listed*

```
/home/cis90/simben $ ls Poems/
ant  Blake  nursery  Shakespeare  twister  Yeats
```

*With a **directory** specified as an argument, the contents of the directory will be listed*

ls command

specifying multiple directories

*The **ls** command can take multiple arguments*

When a file is specified, just the filename is listed

When a directory is specified, the contents of the directory are listed

```
/home/cis90/simben $ ls Poems/ bin/ letter
letter

bin/:
app  banner  enlightenment  hi  I  treed  tryme  zoom

Poems/:
ant  Blake  nursery  Shakespeare  twister  Yeats
```

regular file

directories

ls command example

*The * is expanded by the shell and replaced with the names of all files and directories in the current directory*

```
/home/cis90/simmsben $ ls *
bigfile  letter   proposal1  proposal3   spellk    text.fxd  what_am_i
empty    mission  proposal2  small_town  text.err  timecal

bin:
app  banner  enlightenment  hi  I  treed  tryme  zoom
ls: Hidden: Permission denied

Lab2.0:
386    A_long_name    file.9        READNAME     this_years_annual_report
afile  annual report  junk.old.bak  sTrAnGeNeSs

Lab2.1:
1.1  filename  junk  letter  more  old  Proposal3  Proposal.old  xyz

Miscellaneous:
better_town  file.dos  fruit  manpage  mystery  salad

Poems:
ant  Blake  nursery  Shakespeare  twister  Yeats
```

Files listed first

Then the contents of each directory are listed

Do you see the error message? ... permission issue (more in future lessons)
Do you see the symbolic link? ... in light blue (more in future lessons)

ls command

directory itself vs. contents of a directory (short listing)

```
/home/cis90/simben $ ls bin
app  banner  enlightenment  hi  I  treed  tryme  zoom
```

The contents of the directory are shown

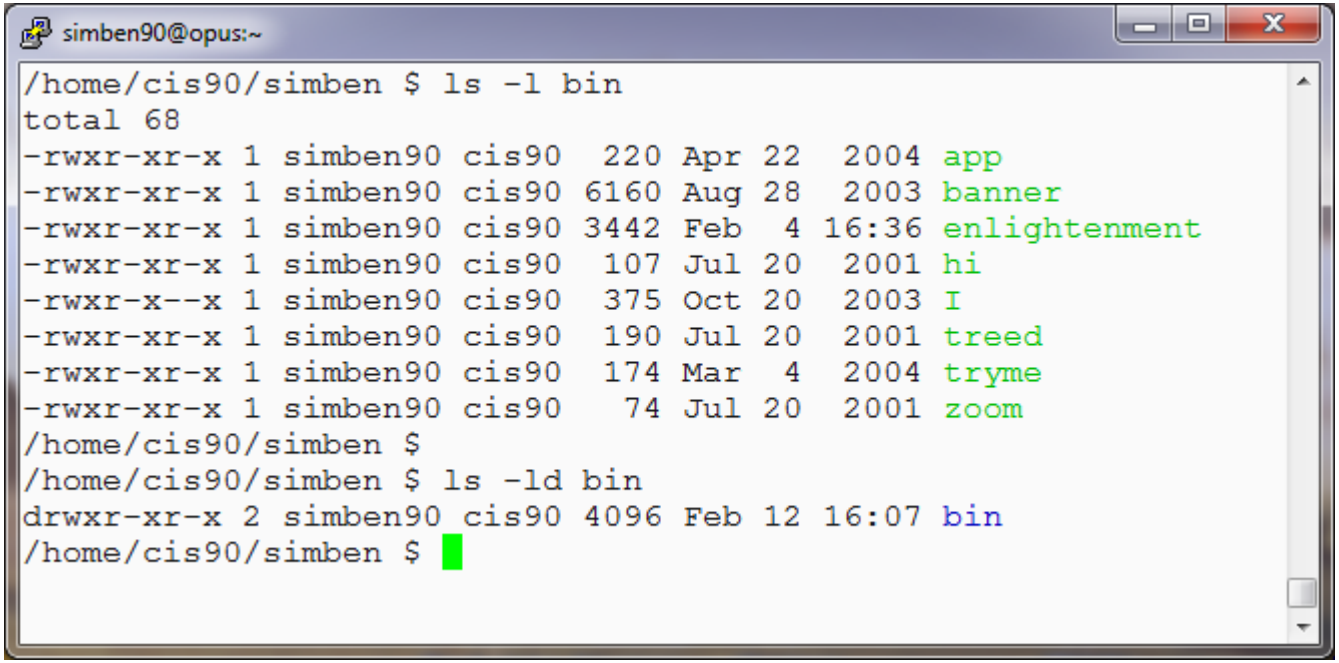
```
/home/cis90/simben $ ls -d bin
bin
```

The directory itself is shown with the -d option

*Use the **d** option to list the directory itself. Without the **d** the directory contents are listed instead.*

ls command

directory itself vs. contents of directory (long listing)



The contents of the directory are shown

The directory itself is shown with the -d option

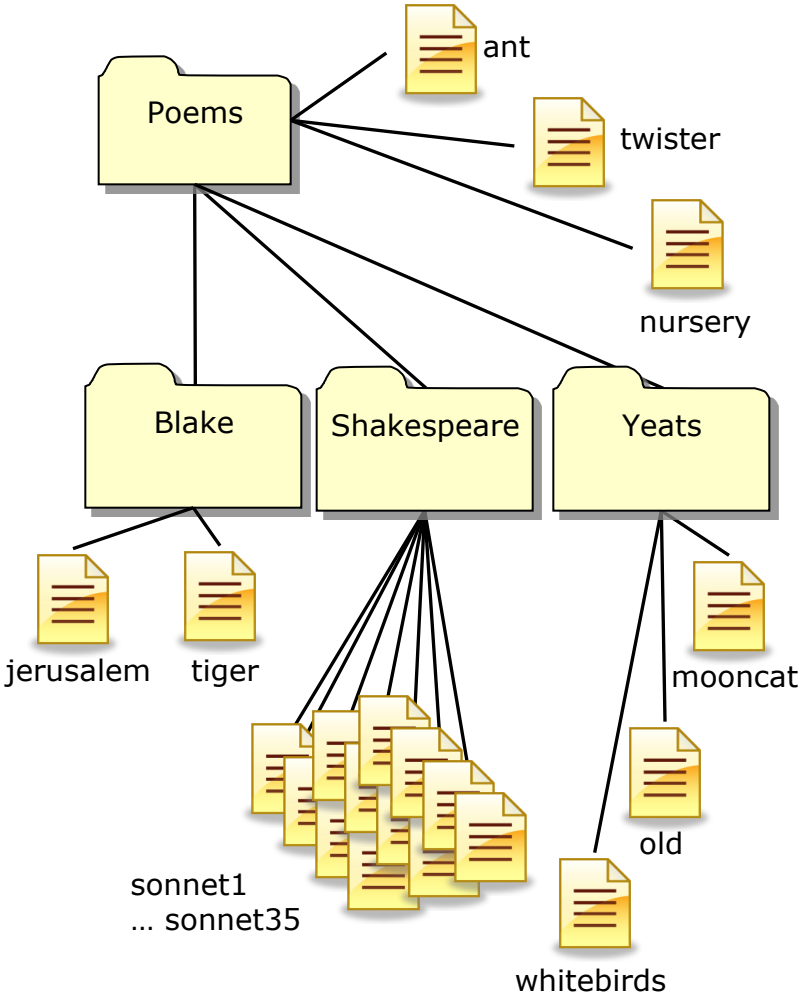
Tip: use the -l and -d options on the ls command to get owner and permission information on directories

ls command

long listing (-l), recursively list subdirectories (-R)

simmsben@opus:~/Poems

```
[simmsben@opus Poems]$ls -lR
.:
total 48
-rw-r--r-- 1 simmsben cis90  237 Aug 26  2003 ant
drwxr-xr-x 2 simmsben cis90 4096 Jul 20  2001 Blake
-rw-r--r-- 1 simmsben cis90  779 Oct 12  2003 nursery
drwxr-xr-x 2 simmsben cis90 4096 Oct 31  2004 Shakespeare
-rw-r--r-- 1 simmsben cis90  151 Jul 20  2001 twister
drwxr-xr-x 2 simmsben cis90 4096 Jul 20  2001 Yeats

./Blake:
total 16
-rw-r--r-- 1 simmsben cis90 582 Jul 20  2001 jerusalem
-rw-r--r-- 1 simmsben cis90 115 Jul 20  2001 tiger

./Shakespeare:
total 104
-rw-r--r-- 1 simmsben cis90 614 Jul 20  2001 sonnet1
-rw-r--r-- 1 simmsben cis90 620 Jul 20  2001 sonnet10
-rw-r--r-- 1 simmsben cis90 689 Oct 31  2004 sonnet11
-rw-r--r-- 1 simmsben cis90 618 Jul 20  2001 sonnet15
-rw-r--r-- 1 simmsben cis90 647 Jul 20  2001 sonnet17
-rw-r--r-- 1 simmsben cis90 631 Jul 20  2001 sonnet2
-rw-r--r-- 1 simmsben cis90 601 Jul 20  2001 sonnet26
-rw-r--r-- 1 simmsben cis90 615 Jul 20  2001 sonnet3
-rw-r--r-- 1 simmsben cis90 598 Jul 20  2001 sonnet35
-rw-r--r-- 1 simmsben cis90 588 Jul 20  2001 sonnet4
-rw-r--r-- 1 simmsben cis90 622 Jul 20  2001 sonnet5
-rw-r--r-- 1 simmsben cis90 581 Jul 20  2001 sonnet7
-rw-r--r-- 1 simmsben cis90 620 Jul 20  2001 sonnet9

./Yeats:
total 24
-rw-r--r-- 1 simmsben cis90 855 Jul 20  2001 mooncat
-rw-r--r-- 1 simmsben cis90 520 Jul 20  2001 old
-rw-r--r-- 1 simmsben cis90 863 Jul 20  2001 whitebirds
[simmsben@opus Poems]$
```



Class Exercise

- Go to your home directory, type: **cd**
- Do a long listing of every file in your home directory and sub-directories and include inode numbers

```
ls –ilR
```

File Types

File Types

Column 1 of long listings shows file types

*Use **ls** with the **-l** option to do a "long listing"*

```
/home/cis90/simben $ ls -l
total 344
-rw-r--r-- 2 simben90 cis90  10576 Jul 20  2001 bigfile
drwxr-xr-x 2 simben90 cis90   4096 Feb 12 16:07 bin
-rw------- 1 simben90 cis90    355 Feb 24 15:40 dead.letter
-rw-r--r-- 1 simben90 cis90      0 Jul 20  2001 empty
d--------- 2 simben90 cis90   4096 Feb  1  2002 Hidden
-r-------- 1 simben90 staff   1182 Feb 16 13:17 lab01.graded
-rw-r--r-- 1 simben90 cis90    494 Feb 12 16:39 lab01-submitted
-r-------- 1 simben90 staff   1873 Feb 23 11:58 lab02.graded
drwxr-xr-x 2 simben90 cis90   4096 Feb 17  2001 Lab2.0
drwxr-xr-x 3 simben90 cis90   4096 Feb 17  2001 Lab2.1
-rw-r--r-- 1 simben90 cis90   1044 Jul 20  2001 letter
-rw-r--r-- 1 simben90 cis90    572 Feb 22 16:07 log
-rw------- 1 simben90 cis90 113773 Feb 28 22:35 mbox
drwxr-xr-x 2 simben90 cis90   4096 Sep 11  2005 Miscellaneous
-rw-r--r-- 1 simben90 cis90    759 Jun  6  2002 mission
drwxr-xr-x 5 simben90 cis90   4096 Jan 18  2004 Poems
-rw-r--r-- 1 simben90 cis90   1074 Aug 26  2003 proposal1
-rw-r--r-- 1 simben90 cis90   2175 Jul 20  2001 proposal2
-rw-r--r-- 1 simben90 cis90   2054 Sep 14  2003 proposal3
-rw-r--r-- 1 simben90 cis90   1580 Nov 16  2004 small_town
-rw-r--r-- 1 simben90 cis90    485 Aug 26  2003 spellk
-rw-r--r-- 1 simben90 cis90    250 Jul 20  2001 text.err
-rw-r--r-- 1 simben90 cis90    231 Jul 20  2001 text.fxd
-rwxr-xr-x 1 simben90 cis90    509 Jun  6  2002 timecal
-rw-rw-r-- 1 simben90 cis90  21295 Feb 28 20:46 uhistory
-rw-r--r-- 1 simben90 cis90    352 Jul 20  2001 what_am_i
/home/cis90/simben $
```

*A "d" indicates a **directory***

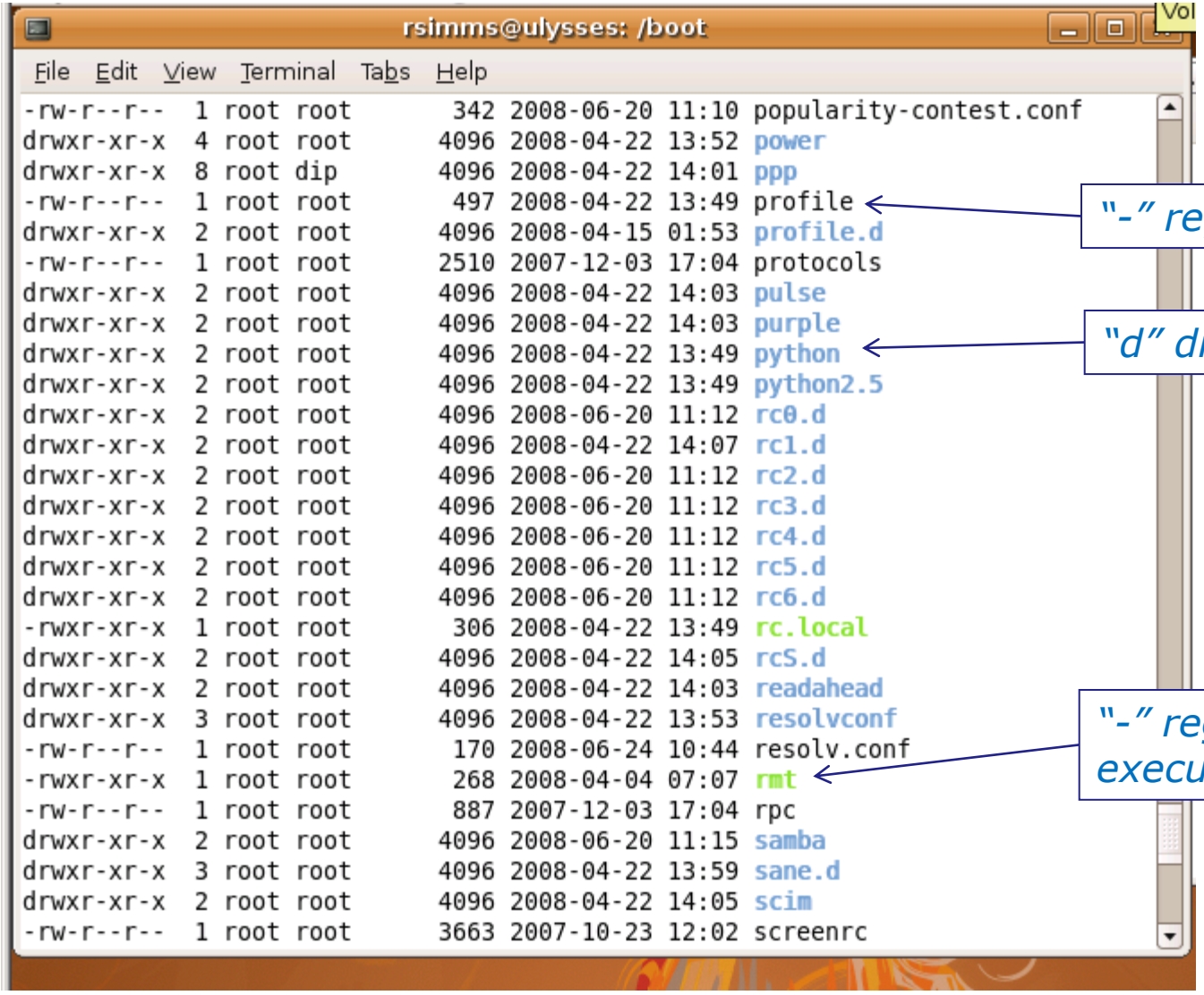
*A "-" indicates a **regular file***

File Types

Column 1 of long listing	Type	How to make one
d	directory	mkdir
-	regular • Programs • Text • Data (binary) *Use the **file** command to classify regular files*	touch vi >
l	symbolic link	ln –s
c	character special device files	mknod
b	block special device files	mknod

Some important types of files

Example File Types
/etc directory on Ubuntu VM

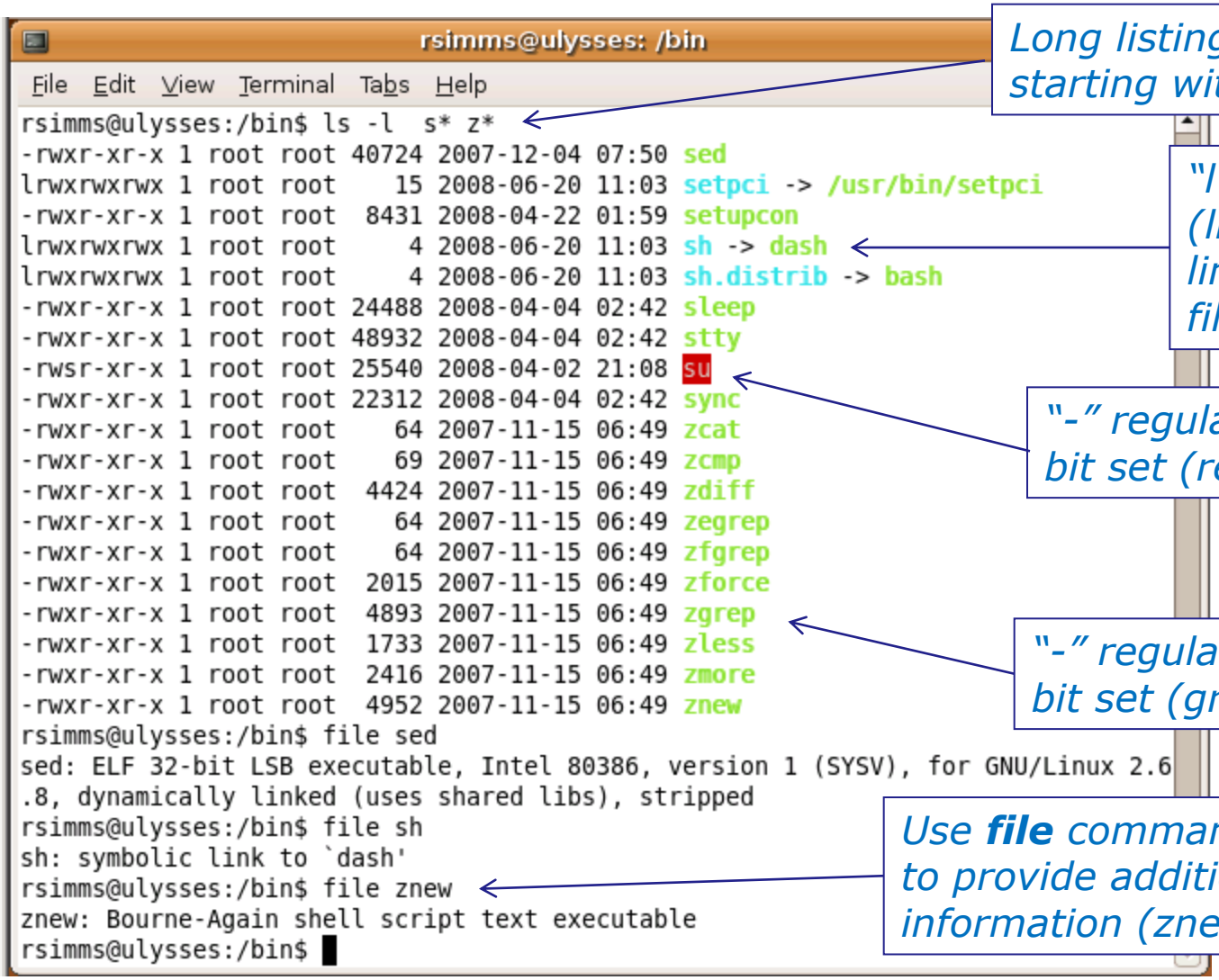


"-" regular file (black)

"d" directory (blue)

"-" regular file with execute bit set (green)

Example File Types
/bin directory on Ubuntu VM



Long listing of files with names starting with s or z

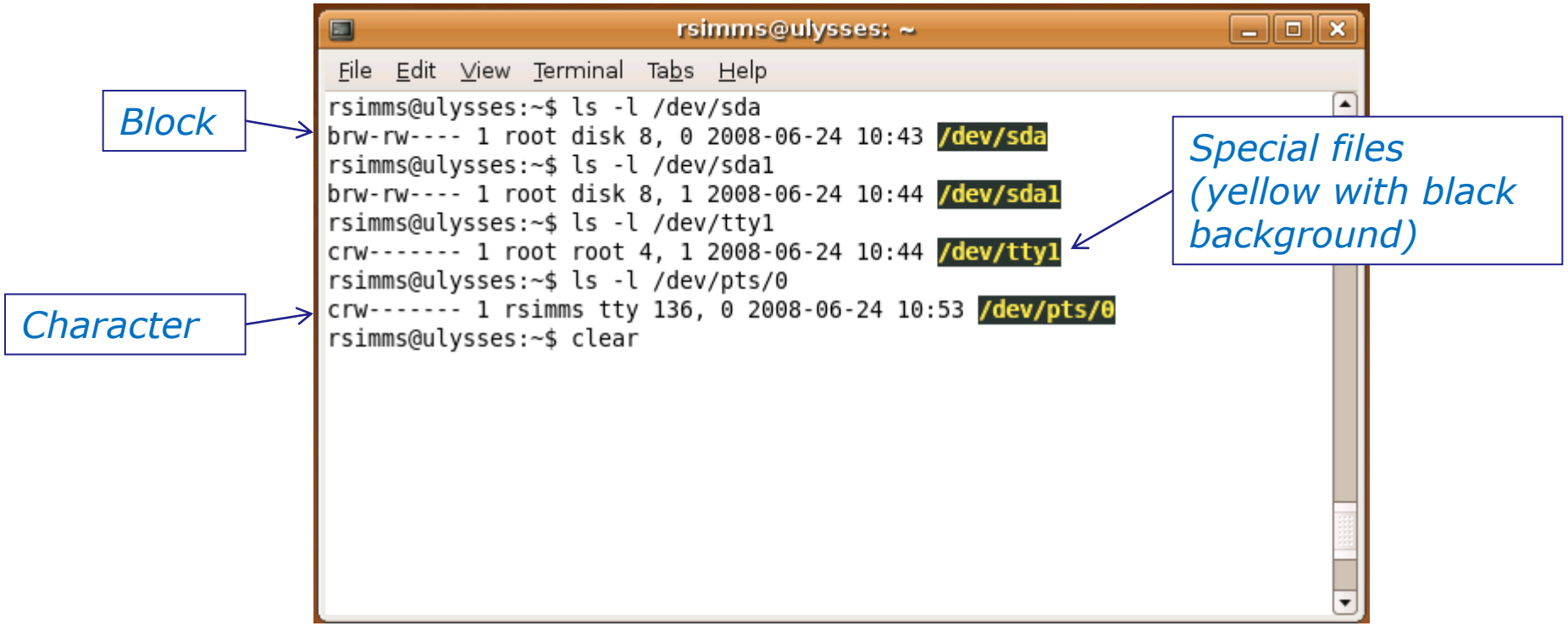
"l" symbolic link (light blue) file sh links to another file named dash

"-" regular file with setuid bit set (red background)

"-" regular file with execute bit set (green)

*Use **file** command to classify a file to provide additional file type information (znew is a shell script)*

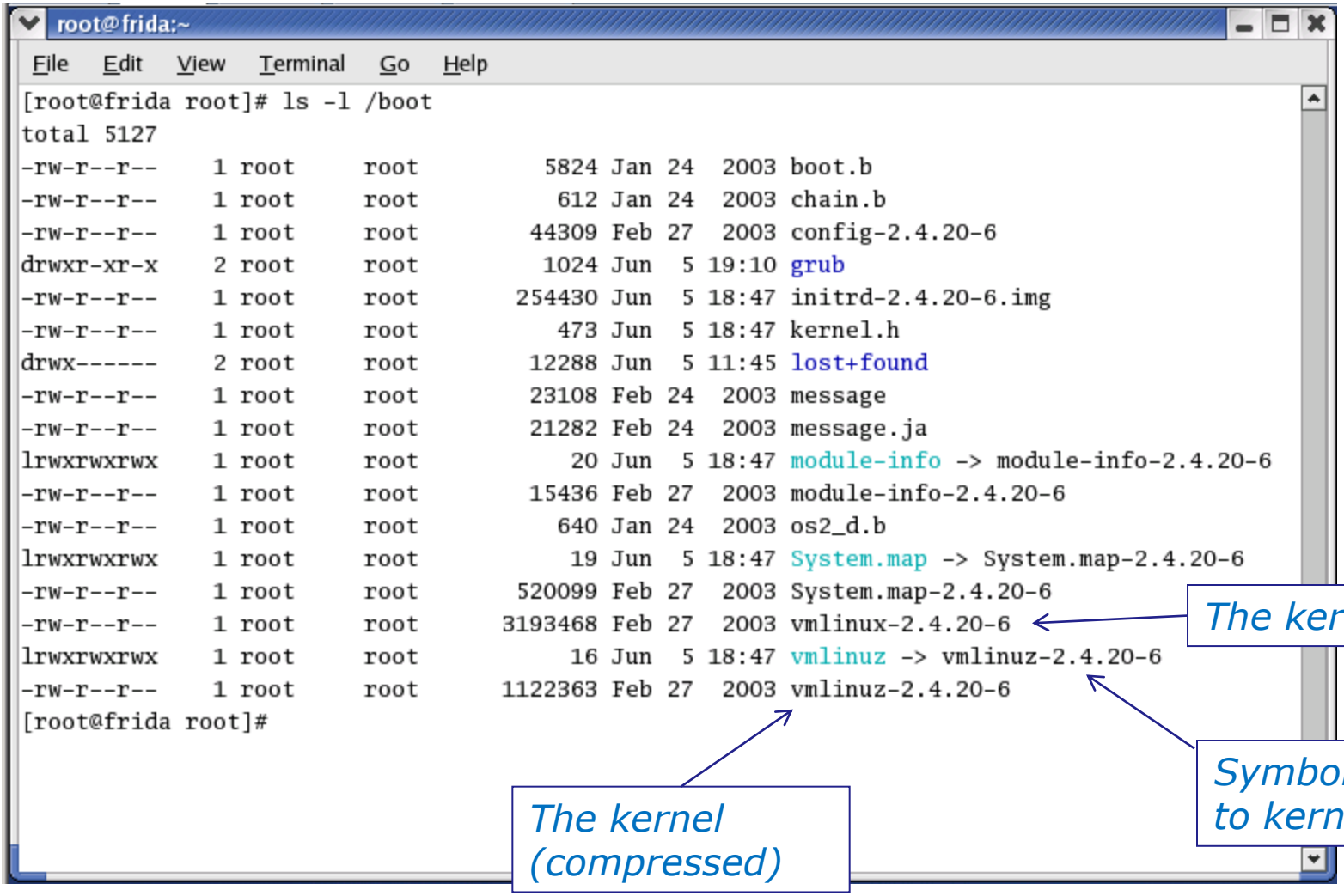
Example File Types
/dev directory on Ubuntu VM



Hard drives are block devices (data is transferred in large chunks for efficiency).

Terminals are character devices where data is transferred one character at a time.

Example File Types
/boot directory on RH9 VM



Class Exercise

Do a long listing of the /etc directory

- Is yum a directory or a regular file?
- What are three ways you can tell if yum is a directory or a file?

UNIX Files
The three elements of a file

```
/home/cis90/simben/Poems $ ls
ant  Blake  nursery  Shakespeare  twister  Yeats

/home/cis90/simben/Poems $ ls -li twister
102625 -rw-r--r-- 1 simben90 cis90 151 Jul 20  2001 twister

/home/cis90/simben/Poems $ cat twister
A tutor who tooted the flute,
tried to tutor two tooters to toot.
Said the two to the tutor,
"is it harder to toot?  Or to
tutor two tooters to toot?"
```

name
+
inode
+
data

Class Exercise
Enlightenment

- **cd** to your home directory on Opus
- Run the enlightenment program: **enlightenment**
- Write down each magic word as you learn them.

Wrap up

Commands:

cat	Print a file on the screen
cd	Change directory
file	Classify a file
head	View first several lines of a file
less	Scroll up and down long files
ls	List files
more	Scroll down long files
pwd	Print working directory
reset	Use to reset terminal window
tail	View last several lines of a file
wc	Count the words, lines or characters in a file
xxd	Hex dump of a binary file

New Files and Directories:

/	Root of the file tree
/home	Opus home directories
/home/cis90	CIS 90 class home directories
/home/cis90/*username*	The home directory for CIS 90 student *username (without the 90)*
/etc/passwd	

Next Class

Assignment: Check Calendar Page on web site to see what is due next week.

Lab 4

Quiz questions for next class:

1) What are two commands you can use to read through long text files?

2) How do you distinguish between relative and absolute paths?

3) What are the three elements of a UNIX file?

Backup

Lab 2 Results

4. Set the TERM environment variable to "dumb", and execute the **clear** command. What does it do? Use **echo $TERM** to see the new setting. Set TERM back to "vt100" or "ansi" What happens?

TERM="dumb"
TERM="ansi"

Set the TERM environment variable back to "xterm" which is what it was when you logged in.

Lab 2 Results

12. What is the difference in output between the following two commands?

banner I am fine
banner "I am fine"