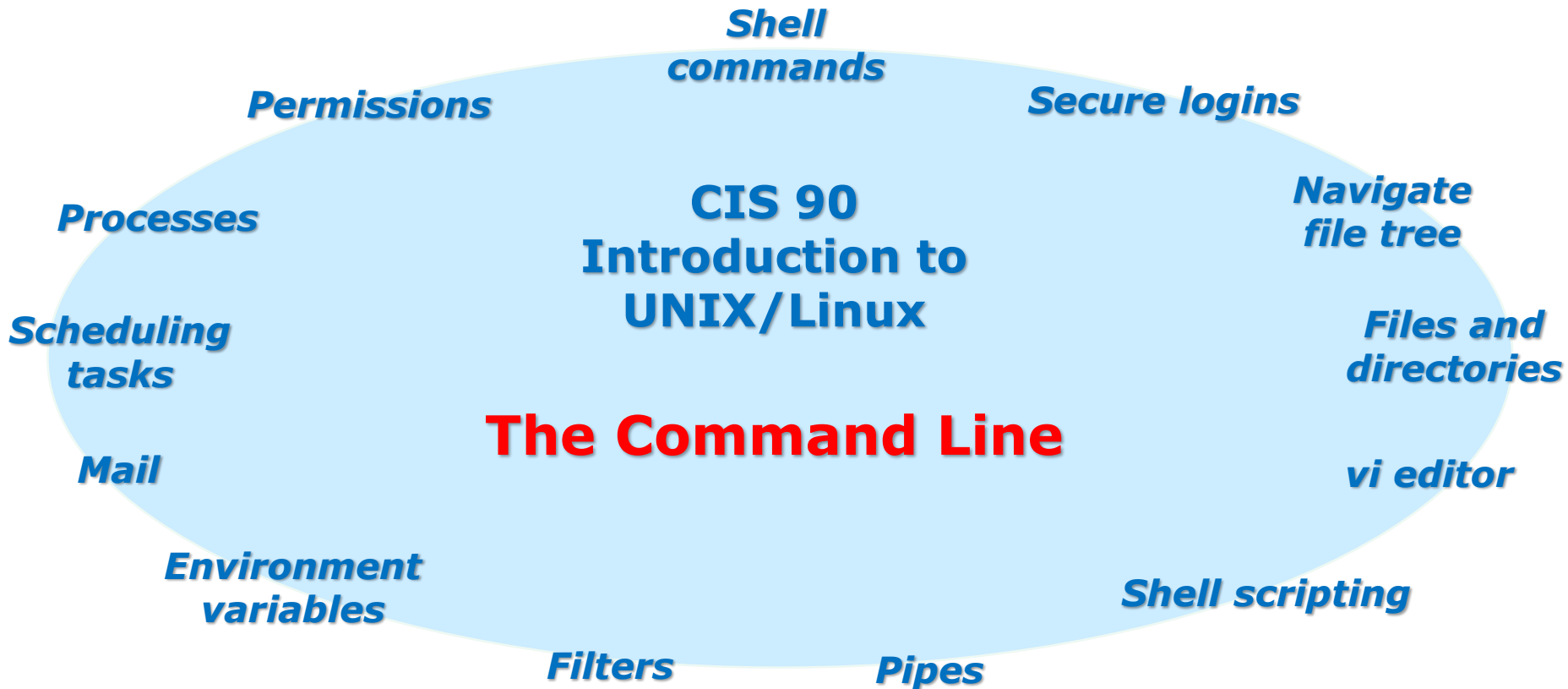




Rich's lesson module checklist

Last updated 02/27/2018

- ☐ Zoom recording named and published for previous lesson
 - ☐ Slides posted
 - ☐ Print out agenda slide and annotate page numbers
 - ☐ 1st minute quiz
 - ☐ Flash cards
 - ☐ Calendar page updated
 - ☐ Sun-Hwa-L5 ready with new accounts and plenty of trouble
 - ☐ Practice test tested (Q16, Q22 and Q30) updated as needed
 - ☐ Canvas test replicated to both sections
 - ☐ Primary and secondary practice test servers up and logins enabled
 - ☐ Q29 email script tested and scheduled to send at end of Lesson 5
 - ☐ Flash cards and timer script ready
 - ☐ Clean up mysql database
 - ☐ Schedule lock of turnin directory and submit
scripts/schedule-submit-locks
 - ☐ 9V backup battery for microphone
 - ☐ Backup slides, CCC info, handouts on flash drive
 - ☐ Key card for classroom door
- ☐ <https://zoom.us>
 - ☐ Putty + Slides + Chrome
 - ☐ Enable/Disable attendee sharing
^ > Advanced Sharing Options > Only Host
 - ☐ Enable/Disable attended annotations
Share > More > Disable Attendee Sharing



Student Learner Outcomes

1. Navigate and manage the UNIX/Linux file system by viewing, copying, moving, renaming, creating, and removing files and directories.
2. Use the UNIX features of file redirection and pipelines to control the flow of data to and from various commands.
3. With the aid of online manual pages, execute UNIX system commands from either a keyboard or a shell script using correct command syntax.

Introductions and Credits



Jim Griffin

- Created this Linux course
- Created Opus and the CIS VLab
- Jim's site: <http://cabrillo.edu/~jgriffin/>



Rich Simms

- HP Alumnus
- Started teaching this course in 2008 when Jim went on sabbatical
- Rich's site: <http://simms-teach.com>

And thanks to:

- John Govsky for many teaching best practices: e.g. the First Minute quizzes, the online forum, and the point grading system (<http://teacherjohn.com/>)



Student checklist - Before class starts

The screenshot shows a web browser window with the URL simms-teach.com/cis90calendar.php. The page title is "Rich's Cabrillo College CIS Classes CIS 90 Calendar". On the left sidebar, there are links for "CIS 90", "CIS 90A", "CIS 90B", "CIS 90C", "CIS 90D", "CIS 90E", "CIS 90F", "CIS 90G", "CIS 90H", "CIS 90I", "CIS 90J", "CIS 90K", "CIS 90L", "CIS 90M", "CIS 90N", "CIS 90O", "CIS 90P", "CIS 90Q", "CIS 90R", "CIS 90S", "CIS 90T", "CIS 90U", "CIS 90V", "CIS 90W", "CIS 90X", "CIS 90Y", "CIS 90Z". The main content area shows the "CIS 90 (Fall 2014) Calendar" with a table of lessons. The "Calendar" link is highlighted. The table has columns for "Lesson", "Date", "Topics", and "Link". The first lesson is "Lesson 1" on "9/2", with topics "Class and Linux Overview" and "Introduction". The "Presentation slides (download)" link is highlighted. The "Enter virtual classroom" link is also highlighted.

Lesson	Date	Topics	Link
Lesson 1	9/2	Class and Linux Overview • Understand how the course will work • High-level overview of computers, operating systems and virtual machines • Overview of UNIX/Linux market and architecture • Using SSH for remote network access • Using terminals and the command line	Presentation slides (download) Enter virtual classroom
Lesson 2	9/9	Supplemental • Homework 1: Logging into Opus (download)	p113-115 p164-172 (pdfs)
Lesson 3	9/16	Assignment • Student Survey • Lab 1	
Lesson 4	9/23	CIS 90 Review	
Lesson 5	9/30	Quiz 1	
Lesson 6	10/7	Commands	

1. Browse to:
<http://simms-teach.com>
2. Click the **CIS 90** link.
3. Click the **Calendar** link.
4. Locate today's lesson.
5. Find the **Presentation slides** for the lesson and **download** for easier viewing.
6. Click the **Enter virtual classroom** link to join ConferZoom.
7. Log into Opus-II with Putty or ssh command.



Student checklist - Before class starts

☐ Google

☐ ConferZoom

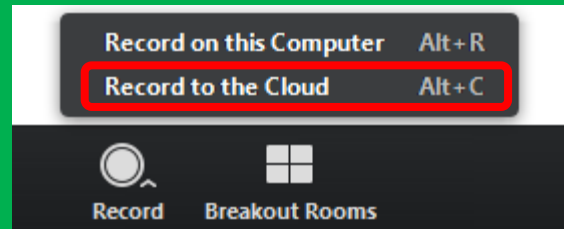
☐ Downloaded PDF of Lesson Slides. I like Foxit Reader so I can take notes using annotations.

The screenshot shows a Zoom meeting interface with several windows open. The main window displays a PDF document titled "Get into the car" with a background image of a white car. Other windows include the Google homepage, the Rich's Cabrillo College CIS 90 website, and a document titled "CIS 90 - Lesson 1" showing a stack of papers and the text "Each student gets their own Arya VM for the term". The Zoom toolbar at the bottom shows options like "Unmute", "Start Video", "Invite", "Participants", "Share Screen", "Chat", "Record", and "Leave Meeting".

☐ CIS 90 website
Calendar page

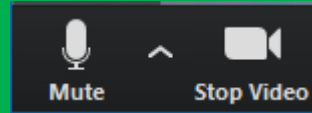
☐ One or more login
sessions to Opus-II

Start



Start Recording

Audio Check



Start Recording

Audio & video Check



Instructor: **Rich Simms**
Dial-in: **408-638-0968 (toll)**
Meeting ID: **426 283 384**



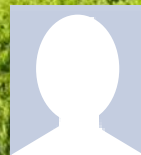
Brandon



Shane



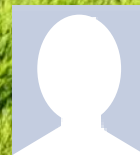
Dan



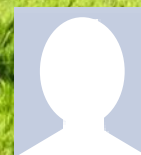
Kage



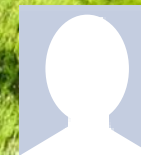
Nathan K.



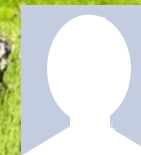
Jo Anne



Darren



Laine



Luis



Christian



Jetta



Cesar



Paul



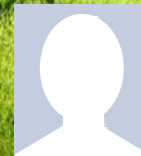
Hilary



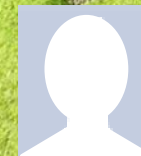
Fritz



Jake



Richard



Nate P.



Ciarán



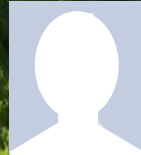
Kimi



November



Henry



Elena



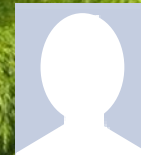
David



Claudius



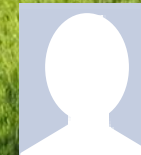
Edgar



Adam

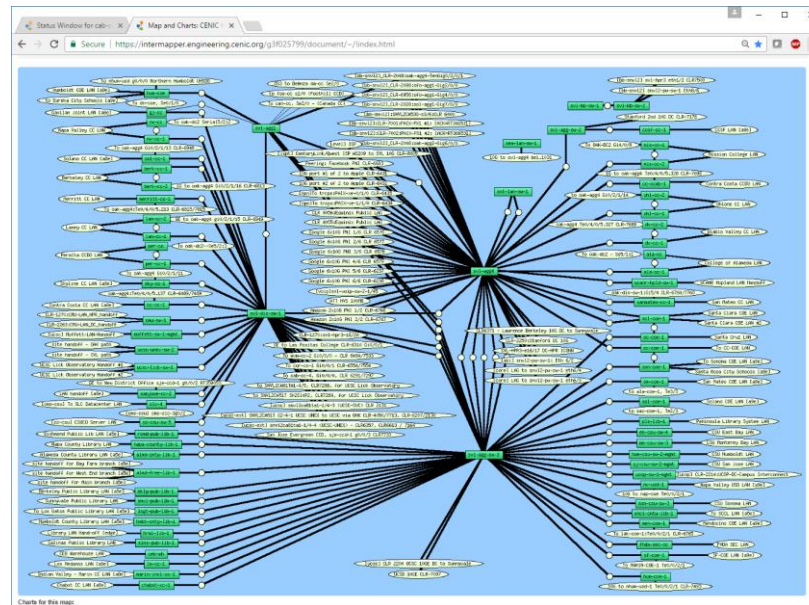


Nathanael T.



Clara

Network Check



[https://intermapper.engineering.cenic.org/g3f025799/
document/~!/index.html](https://intermapper.engineering.cenic.org/g3f025799/document/~!/index.html)

First Minute Quiz

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

Use CCC Confer White Board

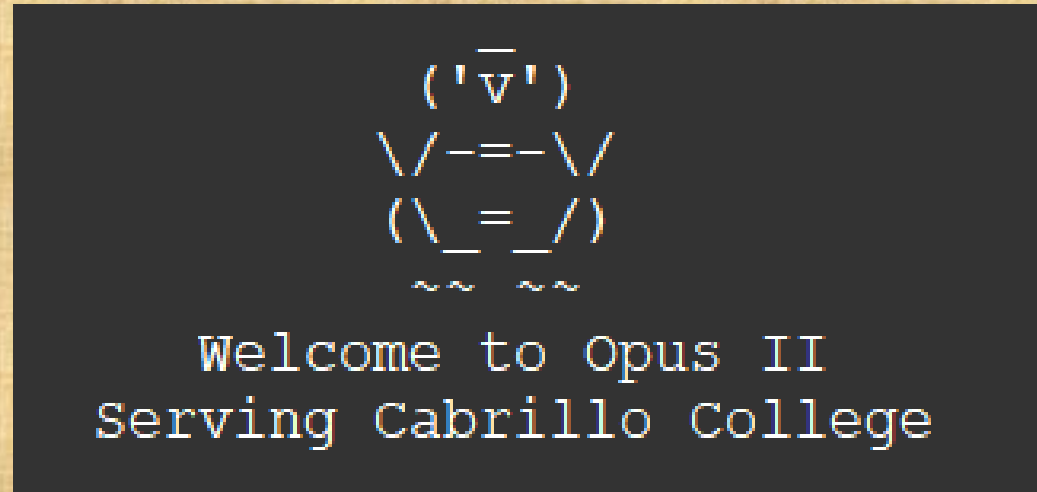
email answers to: risimms@cabrillo.edu

(answers must be emailed within the first few minutes of class for credit)

Review

Objectives	Agenda
• Review Lessons 1-4 • Practice skills • Learn about filename expansion characters	• Quiz • Questions • Six steps of shell (review) • Trouble on the island • Housekeeping • Everything is a file • Filename expansion (globbing) • Filename expansion practice • Command review • Command line syntax & parsing (review) • Command line syntax & parsing practice • Metacharacters (review) • Environment variables (review) • Inputs & outputs (continuing) • The kernel • File system (review) • CCC Confer • Flashcards • Test tips • Assignment • Wrap up

Class Activity



If you haven't already,
log into Opus-II

Class Activity

3	2/19	Unit 3 Electronic Mail • Guest speaker: Denise Moore on OTC (On-The-Job) training programs • Learn how to use the LARC communication tools: write and /bin/mail • Overview on and-to and mail Materials • Presentation slides (download) Supplemental • Howto #318: Accessing vlab (download) Assignment • Read/skim Lesson 3 slides	Lab 2
---	------	--	-----------------------

<https://simms-teach.com/cis90calendar.php>

If you haven't already,
download the lesson slides

Class Activity

	• <u>Read/skim Lesson 1 slides</u> • <u>Student Survey</u> • <u>Lab 1</u>	
	ConferZoom • <u>Enter virtual classroom</u> • <u>Class archives</u>	
	Quiz 1 Commenda • Understand how the UNIX login operation	

<https://simms-teach.com/cis90calendar.php>

If you haven't already, join ConferZoom classroom



Questions

Questions

Lesson material?

Labs?

How this course works?

Are you enlightened yet?



Chinese
Proverb

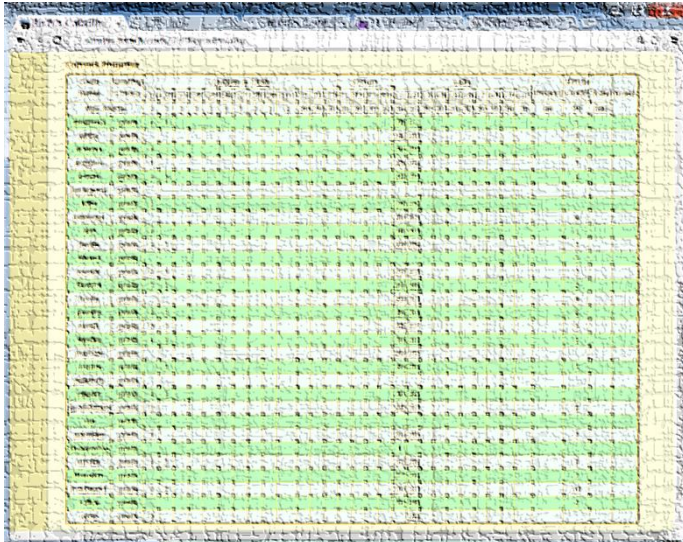
他問一個問題，五分鐘是個傻子，他不問一個問題仍然是一個傻瓜永遠。

He who asks a question is a fool for five minutes; he who does not ask a question remains a fool forever.

Where to find your grades

Send me your survey to get your LOR code name.

The CIS 90 website



<http://simms-teach.com/cis90grades.php>

Percentage	Total Points	Letter Grade	Pass/No Pass
90% or higher	504 or higher	A	Pass
80% to 89.9%	448 to 503	B	Pass
70% to 79.9%	392 to 447	C	Pass
60% to 69.9%	336 to 391	D	No pass
0% to 59.9%	0 to 335	F	No pass

At the end of the term I'll add up all your points and assign you a grade using this table

On Opus-II

checkgrades *codename*
(where codename is your LOR codename)



Written by Jesse Warren a past CIS 90 Alumnus

Points that could have been earned to date:

3 quizzes: 9 points
3 labs: 90 points
1 forum quarter: 20 points
Total: 119 points

Extra Credit

On the forum

Be sure to monitor the forum as I may post extra credit opportunities without any other notice!

On some labs

Extra credit (2 points)

For a small taste of what you would learn in CIS 191 let's add a new user to your Arya VM. Once added we will see how the new account is represented in `/etc/passwd` and `/etc/shadow`.

1. Log into your Arya VM as the cis90 user. Make sure it's your VM and not someone else's.
2. Install the latest updates:
`sudo apt-get update`
`sudo apt-get upgrade`
3. Add a new user account for yourself. You may make whatever username you wish. The example below shows how Benji would make the same username he uses on Opus:
`sudo useradd -G sudo -c "Benji Simms" -m -s /bin/bash simben90`

In lesson slides (search for extra credit)



On the website

<http://simms-teach.com/cis90grades.php>

For some flexibility, personal preferences or family emergencies there is an additional 90 points available of extra credit activities.

<http://simms-teach.com/cis90extracredit.php>

• **Website content review** - The first person to email the instructor pointing out an error or typo on this website will get one point of extra credit for each unique error. The email must specify the specific document or web page, pinpoint the location of the error, and specify what the correction should be. Duplicate errors count as a single point. This does not apply to pre-published material that has been updated but not yet presented in class. (Up to 20 points total)

Getting Help When Stuck on a Lab Assignment

- Google the topic/error message.
- Search the Lesson Slides (they are PDFs) for a relevant example on how to do something.
- Post a question on the forum. Explain what you are trying to do and what you have tried so far.
- Talk to a STEM center tutor/assistant.
- Come see me during my office or lab hours. **I will be in the CTC (room 1403) every Wednesday afternoon from 3-5:30.**
- Make use of the Open Questions time at the start of every class.
- Make a cheat sheet of commands and examples so you never again get stuck on the same thing!

Expect to do a LOT of troubleshooting in this course!

Help Available in the CIS Lab

Instructors, lab assistants and equipment are available for CIS students to work on assignments.



Rich's Cabrillo College CIS Classes
Home Page

Home

Resources

Forums

CIS Lab

Canvas

CIS Lab

webhawks.org/~cislab/

CIS Lab & Datacenter
Aptos Campus

Home Resources NETLAB VLab Location

Announcements

The CIS Lab is in the **STEM Center** in building 800.
A great place to work on lab assignments and get help from student lab assistants and instructors on the schedule below.

STEM CIS/CS hours

Today Jan 28 - Feb 3, 2018 Week

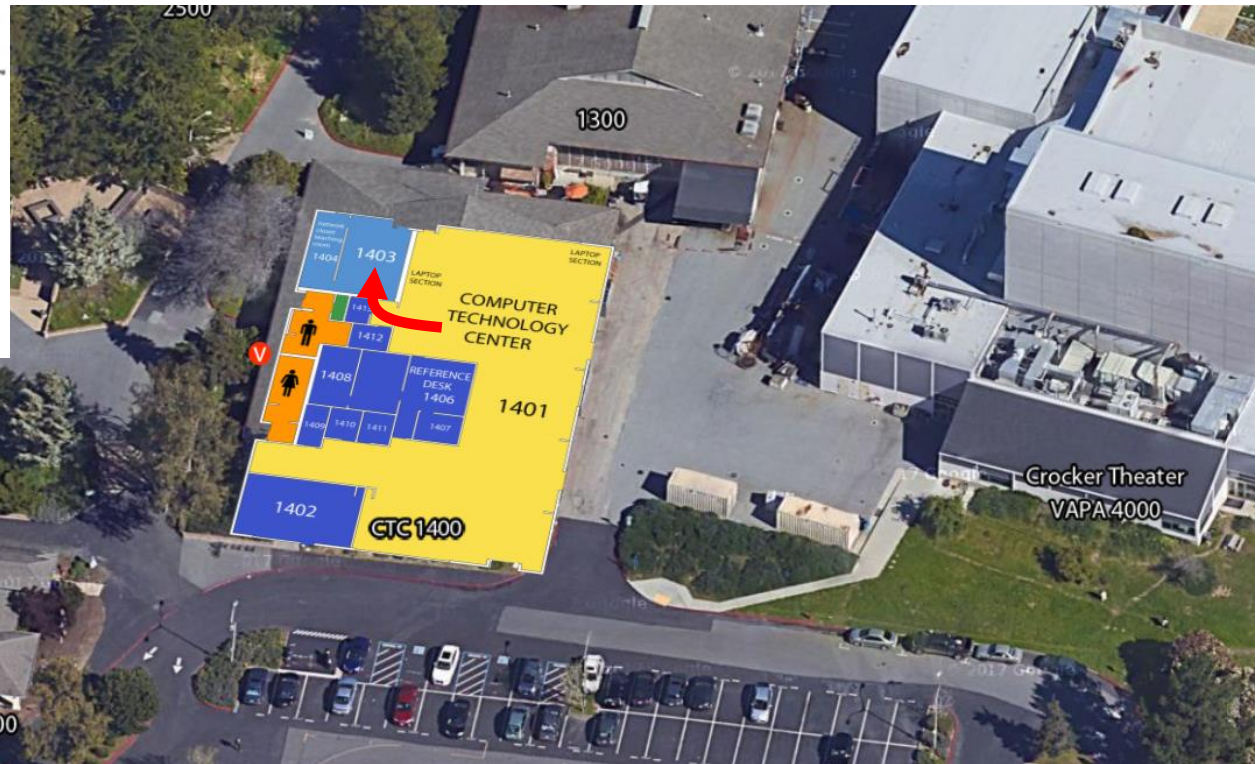
Time	Sun 1/28	Mon 1/29	Tue 1/30	Wed 1/31	Thu 2/1	Fri 2/2	Sat 2/3
10am							
11am							
12pm							
1pm							
2pm		Jeffrey Bergamini CS Instructor 2:10p - 3p Carter Frost CIS/CS	Jeffrey Bergamini CS Instructor 2:10p - 3p Carter Frost CIS/CS	Jeffrey Bergamini CS Instructor 2:10p - 3p Carter Frost CIS/CS	Jeffrey Bergamini CS Instructor 2:10p - 3p Carter Frost CIS/CS		
3pm							
4pm							
5pm							
6pm							
7pm							

Events shown in time zone: Pacific Time

W3C XHTML 1.0 W3C CSS

To see schedule, click the CIS Lab link on the website and use the "Week" calendar view

CTC - Building 1400 On lower campus



I will be in the CTC (room 1403) every Wednesday
afternoon from 3-5:30



The slippery slope



- 1) If you didn't submit the last lab ...
- 2) If you were in class and didn't submit the last quiz ...
- 3) If you didn't send me the student survey assigned in Lesson 1 ...
- 4) If you haven't made a forum post in the last quarter of the course ...

*Please contact me by email, see me during
my office hours or when I'm in the CTC*

Email: risimms@cabrillo.edu



Six Steps of the shell

(review)

Which shell are you using?

```
/home/cis90/simben/Poems/Yeats $ ls /bin/{?,??}sh  
/bin/bash /bin/chsh /bin/csh /bin/ksh /bin/rksh /bin/ssh /bin/tcsh
```

```
/home/cis90/simben/Poems/Yeats $ grep simben90 /etc/passwd  
simben90:x:1201:190:Benji Simms:/home/cis90/simben:/bin/bash
```

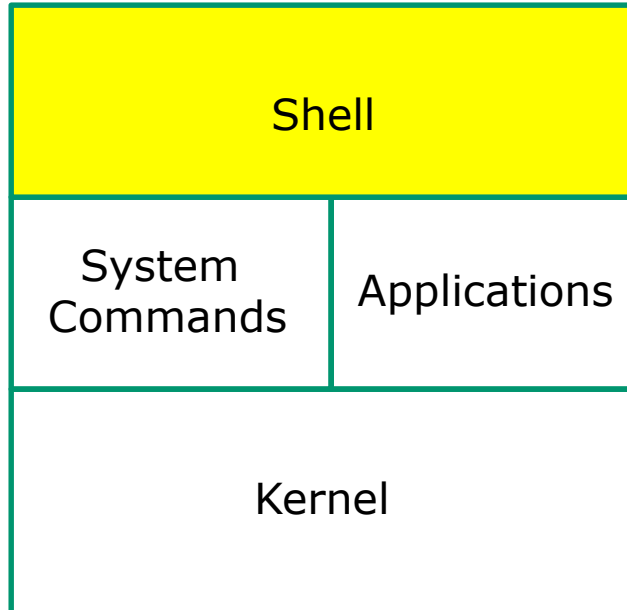
```
/home/cis90/simben/Poems/Yeats $ ps  
  PID TTY          TIME CMD  
 4635 pts/0    00:00:00 bash  
 4785 pts/0    00:00:00 ps
```

```
/home/cis90/simben/Poems/Yeats $ echo $SHELL  
/bin/bash
```

There are many shells on Opus-II. They can be found in the /bin directory. Your account entry in /etc/passwd determines which shell you will use.



Life of the Shell



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



An example of the shell and a command working together as a team to get things done

```

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

output from command

Step	Bash shell /bin/bash	File command /usr/bin/file
Prompt	✓	
Parse	✓	
Search	✓	
Execute	✓	✓
Nap		✓
Repeat	✓	

This table indicates for each step whether bash and/or the command is running

1) Prompt - the shell prompts user for a command

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

Every time you hit the Enter key the shell will prompt you for another command

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



1) Prompt - the shell prompts user for a command

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

```
/home/cis90/simben/Poems/Yeats $ echo $PS1
$PWD $
_ _
```

The shell uses the value of the PS1 variable to make the prompt.

```
/home/cis90/simben/Poems/Yeats $ echo $PWD $
/home/cis90/simben/Poems/Yeats $
```

You can manually do the same thing by echoing the value of the PS1 variable.

FYI, your PS1 variable on Opus-II gets set when you login via a login script containing this command: `PS1='$PWD $ '`

That makes your prompt string be the value of the PWD variable followed by a dollar sign followed by a space. The PWD variable always shows where you are in the UNIX file tree.

2) Parse - the shell parses what you entered

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

The shell parses what you entered and identifies the command, the options, the arguments and any redirection

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

2) Parse - the shell parses what you entered

**Change to your Poems/Yeats directory and
parse this command:**

file *

Command:

Options:

Number of arguments:

Arguments:

Redirection:

Put your answers in the chat window

2) Parse - the shell parses what you entered

**Change to your Poems/Yeats directory and
parse this command:**

file *

Command: `file`

Options: `na`

Number of arguments: `3`

Arguments: `mooncat old whitebirds`

Redirection: `na`

2) Parse - the shell parses what you entered

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

file *

How many arguments is this?

*Use the **echo** command to find out*

```
/home/cis90/simben/Poems/Yeats $ echo *  
mooncat old whitebirds
```

There are actually three arguments!

3) Search - the shell searches the path for your command

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

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

*The shell searches
your path for the
command you entered.*

Where does the shell find the command?

You can do this yourself manually as follows:

```
/home/cis90/simben/Poems/Yeats $ type file  
file is /usr/bin/file
```

```
/home/cis90/simben/Poems/Yeats $ echo $PATH
```

```

1      2      2
/usr/local/bin: /usr/bin: /usr/local/sbin:
4      5
/usr/sbin: /home/cis90/simben/../bin:
6      7
/home/cis90/simben/bin:.
```

*The **file** command is in
the /usr/bin directory.*

*Note: On Centos 7 the
/bin directory is
symbolically linked to the
/usr/bin/ directory.*

*The /usr/bin directory is
the 2nd directory on your
path. Each directory is
delimited by ":"
characters.*

4) Execute - the shell executes the command program file

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

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

*The next step is to load the **file** command that was found into memory. The program on the hard drive becomes a **process** in memory with a unique PID (Process ID).*

*Each new process is given three file descriptors **stdin**, **stdout** and **stderr** for input and output purposes.*

*These are sometimes referred to as the three **standard IO (Input/Output) streams**.*

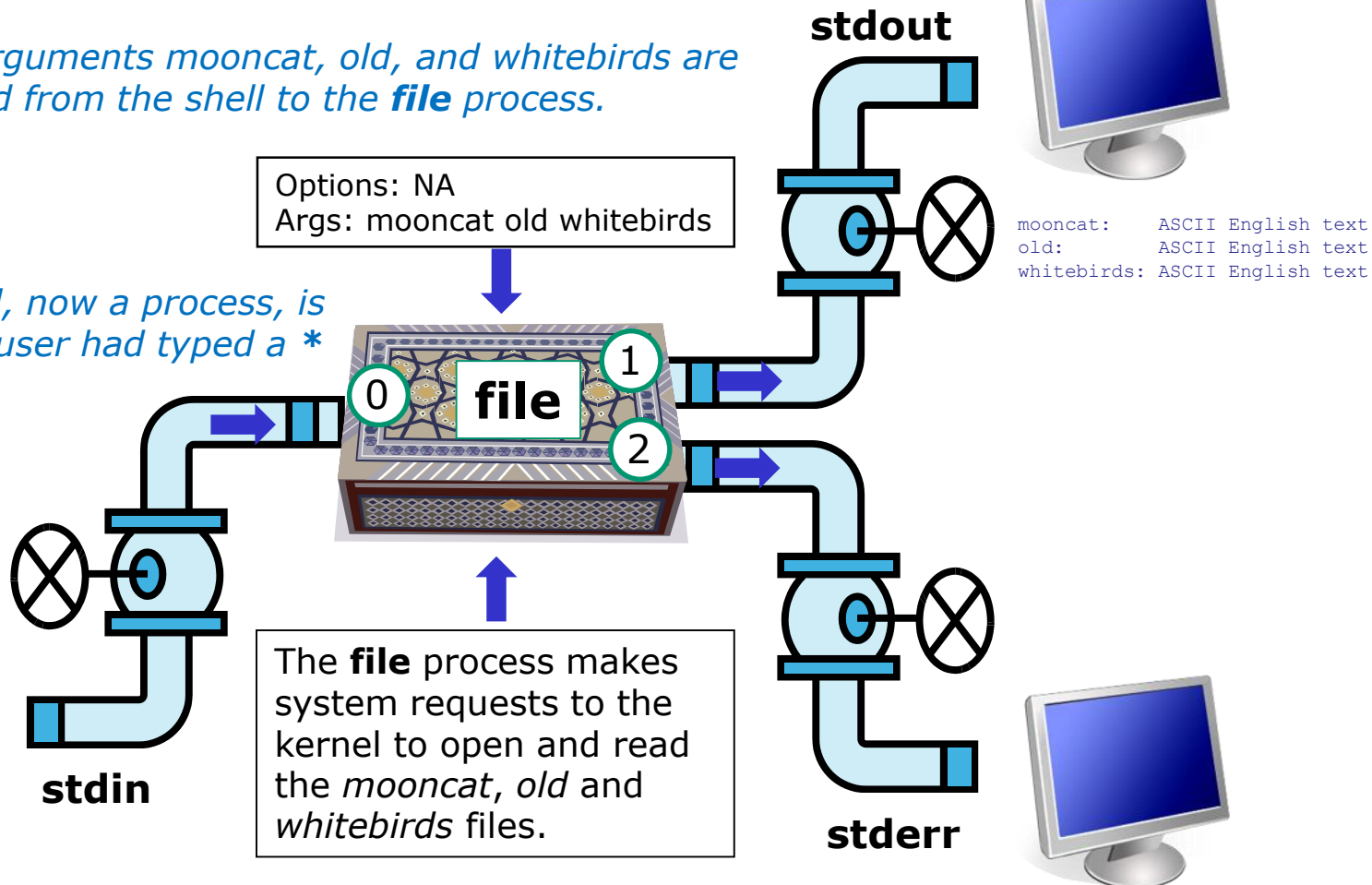
4) Execute - the command is run

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

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

*The arguments mooncat, old, and whitebirds are passed from the shell to the **file** process.*

*The **file** command, now a process, is unaware that the user had typed a **



5) Nap - the shell sleeps while the command runs

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

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

```
mooncat: ASCII English text
```

```
old: ASCII English text
```

```
whitebirds: ASCII English text
```



Output from the file command

The shell's nap ends when the file command has finished

6) Repeat - the shell does it again

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

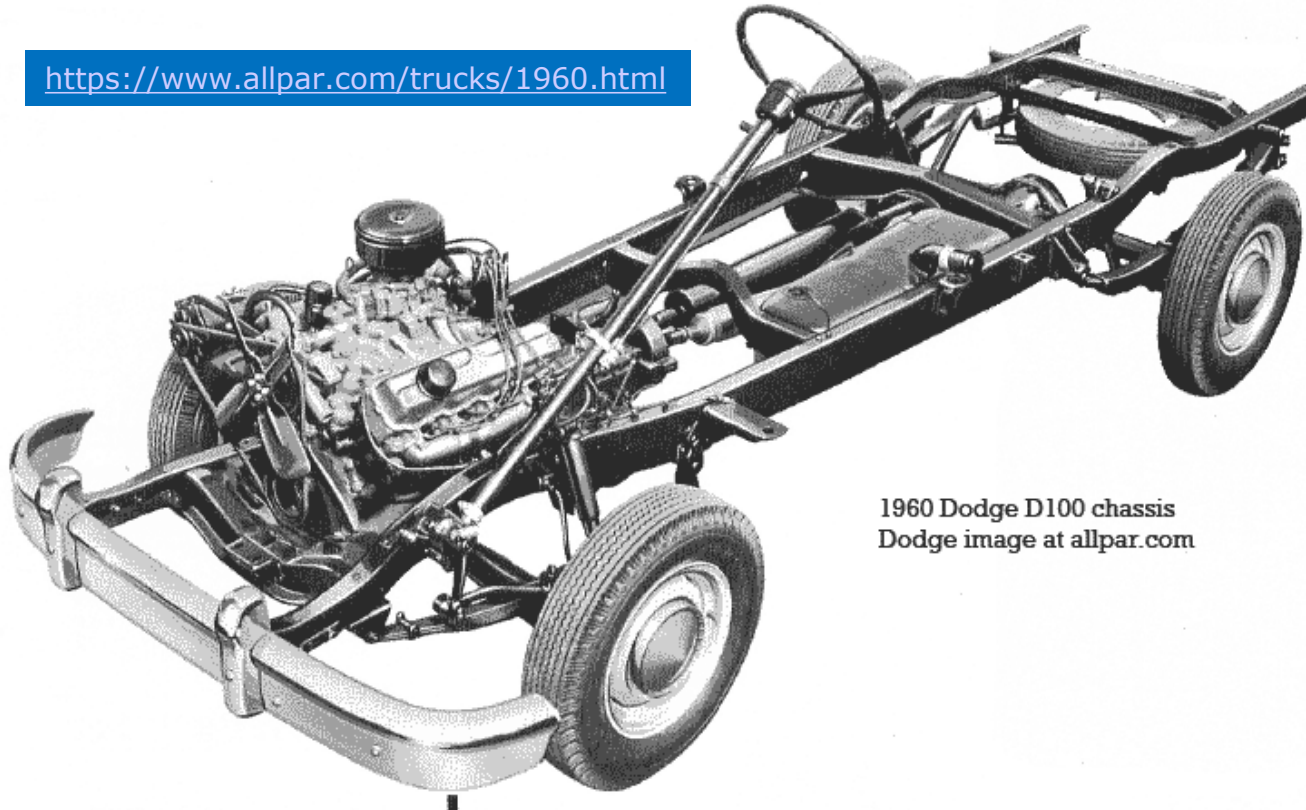
And then it does it all over again for the next command



Some things
are just
important!

Having tires on your car is important

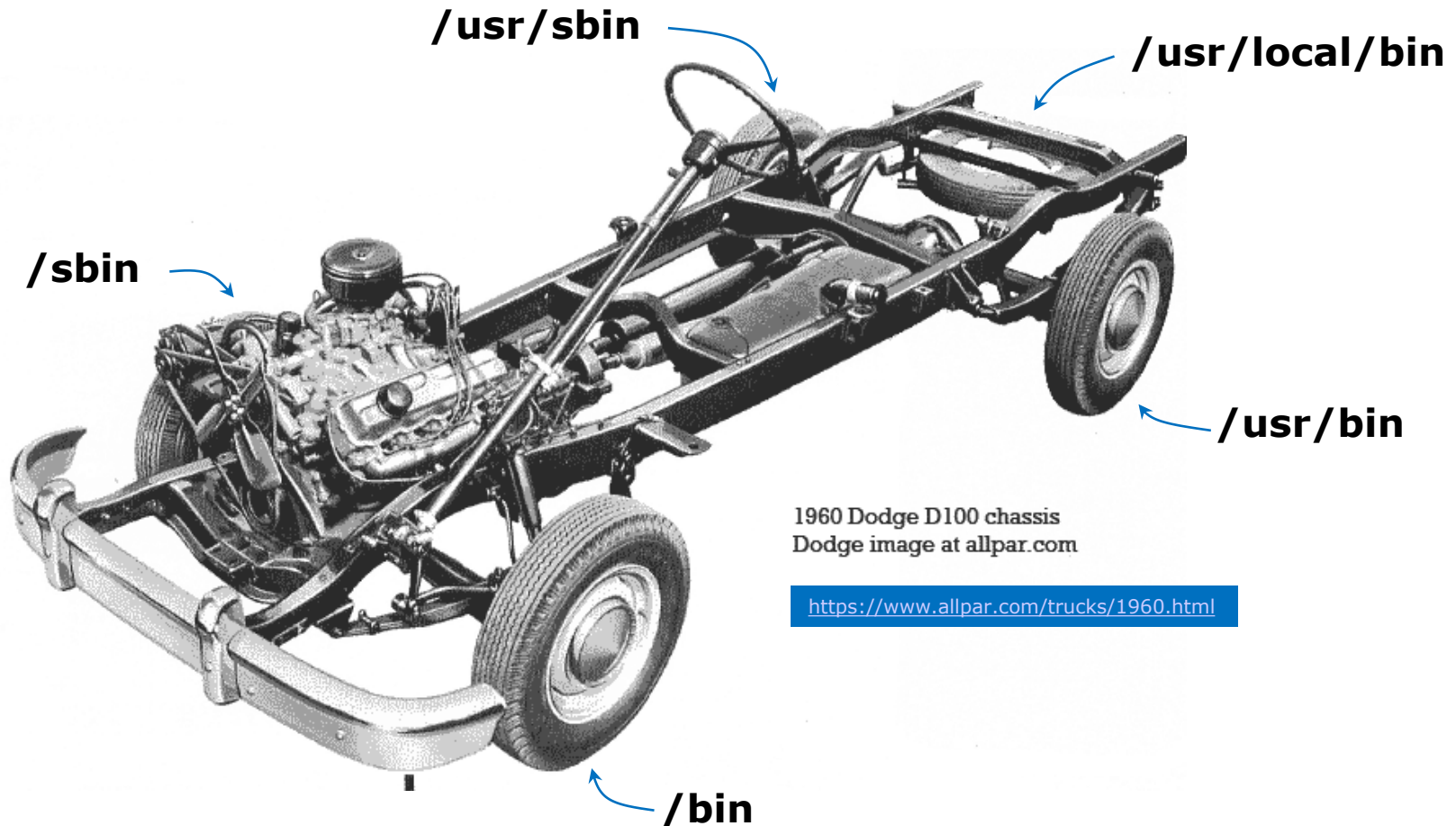
<https://www.allpar.com/trucks/1960.html>



1960 Dodge D100 chassis
Dodge image at allpar.com

How many tires can you count?

Having directories on your path is important



From this day on I want you to be just as concerned if you see one of these directories missing from your path as you would be if one of the tires was missing from your car!

Some excellent directories to have on your path for this course:

/bin

/usr/bin

/usr/local/bin

/sbin

/usr/sbin

Wouldn't hurt to just memorize them. Makes it a lot easier on future CIS 90 tests!

Do you like this path?

```
[simben90@somewhere ~]$ echo $PATH  
/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Put in the chat window what you don't like about it

Could you fix this path?

```
[simben90@somewhere ~]$ echo $PATH  
/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Put in the chat window how you would fix it

Do you like this path?

```
[simben90@somewhere ~]$ echo $PATH  
/etc/trouble:/bin:/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Put in the chat window what you don't like about it

Could you fix this path?

```
[simben90@somewhere ~]$ echo $PATH  
/etc/trouble:/bin:/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Put in the chat window how you would fix it

Answers

```
$ echo $PATH
```

```
/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Oh no ... missing /bin and /usr/bin!

To fix, reset PATH using copy and paste and editing in missing directories

```
$ PATH=/bin:/usr/bin:/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

A simpler way would be to do this

```
$ PATH=/bin:/usr/bin:$PATH
```


Answers

```
$ echo $PATH
```

```
/etc/trouble:/bin:/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

First directory looks suspicious and /usr/bin is missing!

To fix, reset PATH using copy and paste, editing out suspicious directories and adding in missing ones.

```
$ PATH=/bin:/usr/bin:/sbin:/usr/sbin:/usr/local/bin:/home/cis90/simben/bin
```

Trouble on the island today



Reminder to instructor:

On Sun-Hwa-vi, run trouble-L5 as root, rm /etc/nologin

*Now that you
know how the
shell works can
you cat a file?*



*Rich needs to enable logins on sun-hwa-L5 before
starting next activity*

Warm-up Activity

From Opus-II, login to Sun-Hwa-L5 as follows:

ssh sun-hwa-L5

or **ssh \$LOGNAME@sun-hwa-L5**

or **ssh \$LOGNAME@sun-hwa-L5.cis.cabrillo.edu**

After logging in, try to **cat** this file: */etc/mensaje*

If successful:

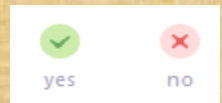
then

Click green "yes" check on ConferZoom Participant window

Help your neighbor

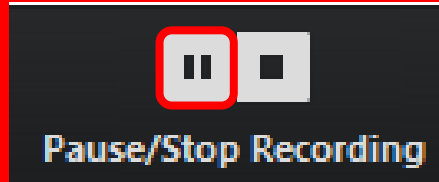
else

Start TROUBLESHOOTING!



Housekeeping





Pause Recording

Audio Check

Roll Call

If you are watching the archived video please email me to let me know you were here.

risimms@cabrillo.edu

Overlap Students

Don't forget to update the Google
Docs Log when watching the
recording



Resume/Stop Recording

Resume Recording

Audio Check

Note taker found!

4	2/21	Quiz 3 The UNIX File System • 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 file types • Learn appropriate commands to view file contents Materials • Presentation slides (download) Supplemental • Notes from our note taker (download) Assignment • Read/skim Lesson 4 slides • Lab 4 ConferZoom • Enter virtual classroom • Class archives	2.1 2.10 2.17 4.1 4.8 4.11-4.12 (Gillay) 24 p659-669 (Hahn)	Lab 3 5 posts
---	------	--	---	---

Thanks you Nathanael T. for sharing your notes!



- Lab 4 is due by 11:59PM tonight (Opus-II time)
- Use the **submit** command to turn in Lab 4
- There is a **check4** script available

Fine print

- Test 1 is next week!



Can you read what it says here?

Test #1 is next week

Practice test available after class



**Test #1 is next
week**

**Practice test
available after
class**

The background of the slide is a reproduction of a painting by J.M.W. Turner titled 'Rain, Steam, and Great Central Railway, 1862'. It depicts a steam locomotive crossing a railway bridge over a river in a hazy, rainy landscape. The painting is characterized by its soft, atmospheric style and visible brushstrokes.

Test #1 is next week

**Practice test
available after class**

Test next week

30 points, plus some extra credit:

- Open book, open notes, open computer
- You must work alone and not help or receive help from others.
- Online timed 60 minute test using Canvas.
- To be taken during the last hour of class.
- Online "archive watching" students that work can take it later in the day but it must be completed by 11:59 PM.

Practice test systems shutdown before real test starts!

Use the forum to discuss practice test questions

- Post if you get stuck on a question
- Post to share tips with others
- Post if you would like a clarification on a test question
- Post your answer to a question to discuss and get feedback from others

Use the forum to arrange study groups

Practice Test 2 Study Group

Locked Search this topic...

3 posts • Page 1 of 1

Practice Test 2 Study Group

by Tess Pritchard » Wed Apr 01, 2015 11:59 am

I know it's late notice, but Mario and I are going to start working through the practice test tomorrow.
Thursdays 1pm in the CIS Lab.
We'd love it if you could join!

Thanks,
Tess

Tess Pritchard

Posts: 30
Joined: Wed Sep 10, 2014
2:15 pm

Example forum post to meet in the STEM center to study for a CIS 90 test

online study group

by Benjamin Correia » Mon Mar 02, 2015 4:22 pm

I wasn't able to make the study group on campus today due to work so I thought id see if anyone would be willing to meet up online through Skype, hangouts or some other online collaborative work space like Docs...

let me know if your interested in spending an hour or two studying tomorrow morning from 9am to 11, I have work from 12pm until around 10:30 so if people would rather work a night I could devote an hour or so after that time.

if you have any ideas for a online study group pleas feel free to add your thoughts, I am having a bit of trouble really understanding a few things we covered in class so I thought id ask my peers for some advice if possible.

-Benji

Benjamin Correia

Posts: 28
Joined: Tue Feb 03, 2015
11:11 am

Example forum post to meet online to study for a CIS 90 test

To get notifications of new forum posts

Subscribe to the forum to get email notifications of new posts

After logging in:

1. Go to the CIS 90 class forum.
2. At the bottom of the page, click the "Subscribe forum" link on the lower left. When subscribed you get email notifications when new posts are made.
3. To unsubscribe, click it again.

[Home](#) < [Board index](#) ☒ [Subscribe forum](#)

*Unsubscribed
looks like this.*

[Home](#) < [Board index](#) ☐ [Unsubscribe forum](#)

*Subscribed
looks like this.*

Perkins/VTEA Survey

The screenshot shows a forum post on the 'Cabrillo College: Computer and Information Systems' forum. The post is titled 'Carl D. Perkins Vocational and Technical Education Act' and was posted by Rich Simms on Tue Sep 22, 2015 at 1:45 pm. The post text explains that the Carl D. Perkins Vocational and Technical Education Act was originally authorized by Congress in 1966, reauthorized in 1990 and again in 2009. It provides federal funding for higher-level career technical education (CTE) in that the United States in order to help the economy. For Cabrillo College, it receives portions of this funding to develop and disseminate through a survey. The survey already completed the grant funds the college's reserves. The survey only needs to be completed once per year by each student. The survey can be completed online through the link provided. Log on to WEBAUTOR at <https://webauto.cabrillo.edu>. Select "ATTENDANCE" Click Menu (near blue bar) Under "Attendance Profile" click on "Student Update Form" Click on the "Update Form" link under "Select the earliest term for which you are registered" and click on the "Update Form" link under "Update Form". Second, click on the "Career Technical Information" link. Answer questions by clicking on the "Yes" or "No" button. You can get details about a question by clicking on the question's name. After answering all questions, click "Submit". There is a "GO BACK" link. The post also includes a link to the survey: <https://webauto.cabrillo.edu>.

This is an important source of funding for Cabrillo College.

*Send me an email stating you completed this Perkins/VTEA survey for **three points extra credit!***

<https://opus-ii.cis.cabrillo.edu/forum/viewtopic.php?f=6&t=349>

Career Technical Information	
Your answers to these questions will help qualify Cabrillo College for Perkins/VTEA grant funds.	
Are you currently receiving benefits from:	
☐ Yes	TANF/CALWORKS
☐ No	
☐ Yes	SSI (Supplemental Security Income)
☐ No	
☐ Yes	GA (General Assistance)
☐ No	
☐ Yes	Does your <u>income</u> qualify you for a fee waiver?
☐ No	
☐ Yes	Are you a single parent with custody of one or more minor children?
☐ No	
☐ Yes	Are you a <u>displaced homemaker</u> attending Cabrillo to develop job skills?
☐ No	
☐ Yes	Have you moved in the preceding 36 months to obtain, or to accompany parents or spouses to obtain, temporary or seasonal employment in agriculture, dairy, or fishing?
☐ No	

LPI Linux Essentials Certificate

Linux Essentials Certificate of Achievement				
Objective	# of Questions	Cabrillo	Urban Penguin	NDG Linux Essentials
Topic 1: The Linux Community and a Career in Open Source				
1.1 Linux Evolution and Popular Operating Systems	2	CIS90 Lesson 1	1.1	Module 1
1.2 Major Open Source Applications	2	CIS90 Lesson 1	1.2	Module 2
1.3 Understanding Open Source Software and Licensing	1	CIS90 Lesson 1	1.3	Module 2
1.4 ICT Skills and Working in Linux	2	not covered	1.4	Module 3
Topic 2: Finding Your Way on a Linux System				
2.1 Command Line Basics	2	CIS90 Lesson 2	2.1	Module 4
2.2 Using the Command Line to Get Help	2	CIS90 Lesson 2	2.2	Module 5
2.3 Using Directories and Listing Files	2	CIS 90 Lesson 4	2.3	Module 6
2.4 Creating, Moving and Deleting Files	2	CIS90 Lesson 5	2.4	Module 6
Topic 3: The Power of the Command Line				
3.1 Archiving Files on the Command Line	2	CIS 90 Lesson 14	3.1	Module 7
3.2 Searching and Extracting Data from Files	4	CIS 90 Lesson 8	3.2	Module 8
3.3 Turning Commands into a Script	4	CIS 90 Lesson 13 & 14	3.3	Module 9
Topic 4: The Linux Operating System				
4.1 Choosing an Operating System	1	not covered	4.1	Module 1
4.2 Understanding Computer Hardware	2	CIS 90 Lesson 1	4.2	Module 10
4.3 Where Data is Stored	3	CIS 90 Lesson 1	4.3	Module 11
4.4 Your Computer on the Network	2	CIS 192	4.4	Module 12
Topic 5: Security and File Permissions				
5.1 Basic Security and Identifying User Types	2	CIS 191	5.1	Module 13
5.2 Creating Users and Groups	2	CIS 191	5.2	Module 14
5.3 Managing File Permissions and Ownership	2	CIS 90 Lesson 7	5.3	Module 15
5.4 Special Directories and Files	1	CIS 90 Lesson 4	5.4	Module 16



The Urban Penguin

LINUX ESSENTIALS

Home
LPI

Welcome to this self study video series of tutorials. These videos can be used in preparing you for the LPI (Linux Professional Institute), Linux Essentials Certification. These materials are meant as a stand-alone learning solution in readiness for your exam and are targeted towards anyone who is aiming for the certification or just wants to know more about what Linux is and what it can offer. The Urban Penguin is an **Approved LPI Training Partner** and we provide both free training via these videos and, if you prefer to work direct with the penguin, then we can offer [online training](#) at a reasonable cost

Objective	Description	Click to Access
Intro	What is LPI Linux Essentials	Click to Access
1.1	Linux evolution and popular operating systems	Click to Access
1.2	Major Open Source applications	Click to Access
1.3	Understanding Open Source Software and licensing	Click to Access
1.4	ICT skills and working with Linux	Click to Access
2.1	Command line basics	Click to Access
2.2	Using the command line to get help	Click to Access
2.3	Using directories and listing files	Click to Access
2.4	Creating, moving and deleting	Click to Access
3.1	Archiving files from the command line	Click to Access
3.2	Searching and extracting data from files	Click to Access
3.3	Turning commands into a script	Click to Access
4.1	Choosing an operating system	Click to Access
4.2	Understanding computer hardware	Click to Access
4.3	Where data is stored	Click to Access
4.4	Your computer on the network	Click to Access
5.1	Basic security and user types	Click to Access
5.2	Creating users and groups	Click to Access
5.3	Manage file permissions and ownership	Click to Access
5.4	Special directories and files	Click to Access

Instructor led and free video based Linux Training

<http://www.theurbanpenguin.com/lpi/le.html>

*No registration, no logging in,
just click and watch the videos*

NDG Linux Essentials via Cisco Networking Academy

The screenshot shows the NDG Linux Essentials course page. The main content area is titled '2.3 Major Open Source Applications'. It includes a paragraph about the Linux kernel and its role in various hardware platforms. Below this, there's a list of categories: Server software, Desktop software, and Tools. A terminal window on the right shows the Ubuntu login screen with the text 'Starting domain name service... bind9', 'Welcome to Ubuntu 12.04.5 LTS (GNU/Linux 3.17.7 x86_64)', and 'This lab has two user accounts (username :: password)'. The terminal also shows the command 'cat /etc/passwd' and its output.

<https://www.netacad.com/>

*Complete course with reading, live VM
and tests.*

*Contact me if you would like a student
account for the NDG Linux Essentials
course.*

Linux Computer Home Loans



<https://docs.google.com/a/cabrillo.edu/spreadsheets/d/1ljwkXZ7BYcCCo3UwqHz0EPm2I3OMSXYrfYv43C2MBc/edit?usp=sharing>

If interested click the Google Docs link above and request access to the sign-up sheet. Based on the number of requests I'll determine how long they can be checked out for.

Everything is a file

Everything is a file in UNIX (even a terminal)

- A terminal
- A file
- A directory
- A hard drive
- A hard drive partition
- A CD
- A partition on a USB flash drive
- Kernel run-time information

*Implemented as
files in UNIX*

Everything is a file in UNIX (even a terminal)

- A terminal *e.g. /dev/pts/2*
- A file *e.g. /home/cis90/simben/letter*
- A directory *e.g. /home/cis90/*
- A hard drive *e.g. /dev/sda*
- A hard drive partition *e.g. /dev/sda1*
- A CD *e.g. /dev/cdrom*
- A partition on a USB flash drive *e.g. /dev/sdb2*
- Kernel run-time information *e.g. /proc/sys/kernel/hostname*

Everything is a file in UNIX (even a terminal)

```
[rsimms@oslab ~]$ ls -l /dev/pts/3
```

```
crw--w----. 1 leebri90 tty 136, 3 Sep 30 16:33 /dev/pts/3
```

terminal

```
[rsimms@oslab ~]$ ls -l /home/cis90/simben/letter
```

```
-rw-r--r--. 1 simben90 cis90 1044 Jul 20 2001 /home/cis90/simben/letter
```

file

```
[rsimms@oslab ~]$ ls -ld /home/cis90/
```

```
drwxr-x---. 43 rsimms cis90 4096 Sep 16 15:00 /home/cis90/
```

directory

```
[rsimms@oslab ~]$ ls -l /dev/sda
```

```
brw-rw----. 1 root disk 8, 0 Sep 13 17:47 /dev/sda
```

hard drive

```
[rsimms@oslab ~]$ ls -l /dev/sda1
```

```
brw-rw----. 1 root disk 8, 1 Sep 13 17:47 /dev/sda1
```

partition

```
[rsimms@oslab ~]$ ls -l /dev/cdrom
```

```
lrwxrwxrwx. 1 root root 3 Sep 13 17:46 /dev/cdrom -> sr0
```

CD drive

```
[rsimms@oslab ~]$ ls -l /dev/sr0
```

```
brw-rw----. 1 root cdrom 11, 0 Sep 13 17:46 /dev/sr0
```

*Kernel
runtime info*

```
[rsimms@oslab ~]$ ls -l /proc/sys/kernel/hostname
```

```
-rw-r--r-- 1 root root 0 Sep 24 15:45 /proc/sys/kernel/hostname
```

File Types

Long listing code (ls -l)	Type	How to make one
d	directory	mkdir
-	Regular file • Programs • Text • Data (binary)	touch
l	symbolic link	ln -s
c	character device file	mknod
b	block device file	mknod

Common file types in a Linux file system

Everything is a file in UNIX (even a terminal)

Nice things about files

- you can write to them

```
[rsimms@opus ~]$ echo "Rich was here" > myfile
```

- and read from them

```
[rsimms@opus ~]$ cat myfile  
Rich was here
```

Class Activity

- Write to a file

```
echo "Rumpelstiltskin was here" > myfile
```

- Read the file

```
cat myfile
```

Did it work?

Click green "Yes" check or red "No" X
on ConferZoom



yes



no

Everything is a file (even a terminal)

```
/home/cis90/simmsben $ tty  
/dev/pts/1
```

*Use the **tty** command to identify the specific terminal device being used*

Note this device is identified using an absolute pathname

Everything is a file (even a terminal)

```
/home/cis90/simmsben $ tty  
/dev/pts/1
```

Show which terminal you are using

```
/home/cis90/simmsben $ who  
simmsben pts/1      2010-09-29 07:38 (dsl-49-64-10-90.dhcp.cruzio.com)  
srecklau pts/2      2010-09-29 06:06 (62.143.60.194)  
rsimms   pts/4      2010-09-29 06:47 (dsl-49-64-10-90.dhcp.cruzio.com)
```

Use who to see who is logged in

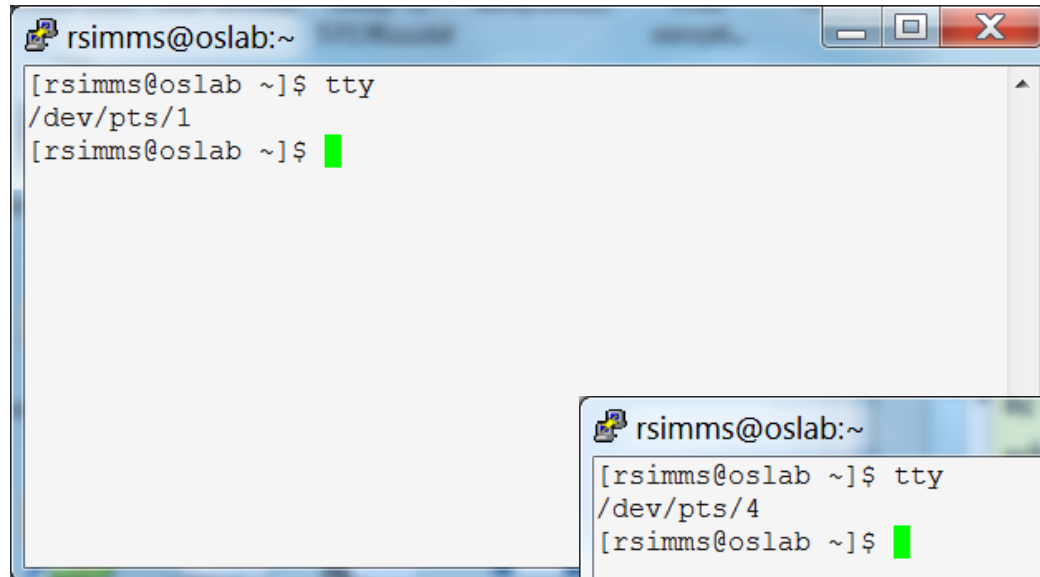
```
/home/cis90/simmsben $ ls -l /dev/pts/*  
crw--w---- 1 simmsben tty 136, 1 Sep 29 07:45 /dev/pts/1  
crw--w---- 1 srecklau tty 136, 2 Sep 29 07:44 /dev/pts/2  
crw--w---- 1 rsimms   tty 136, 4 Sep 29 06:48 /dev/pts/4
```

*Do a long listing to see
all the terminal devices
in use*

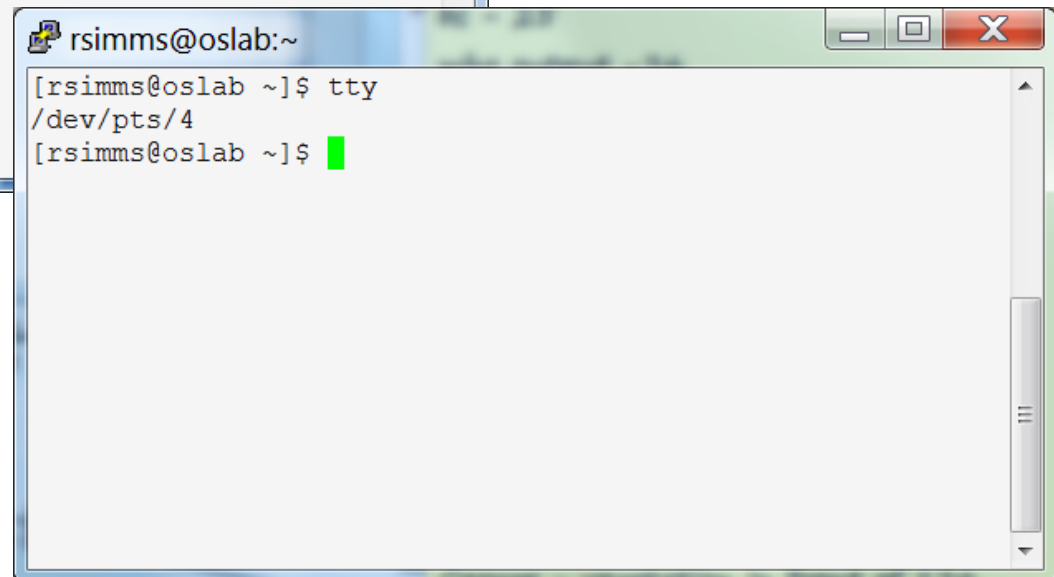
Notice the owner is someone who has logged in

Notice the file type is "c" which is a character device file

Everything is a file (even a terminal)



```
rsimms@oslab:~  
[rsimms@oslab ~]$ tty  
/dev/pts/1  
[rsimms@oslab ~]$
```

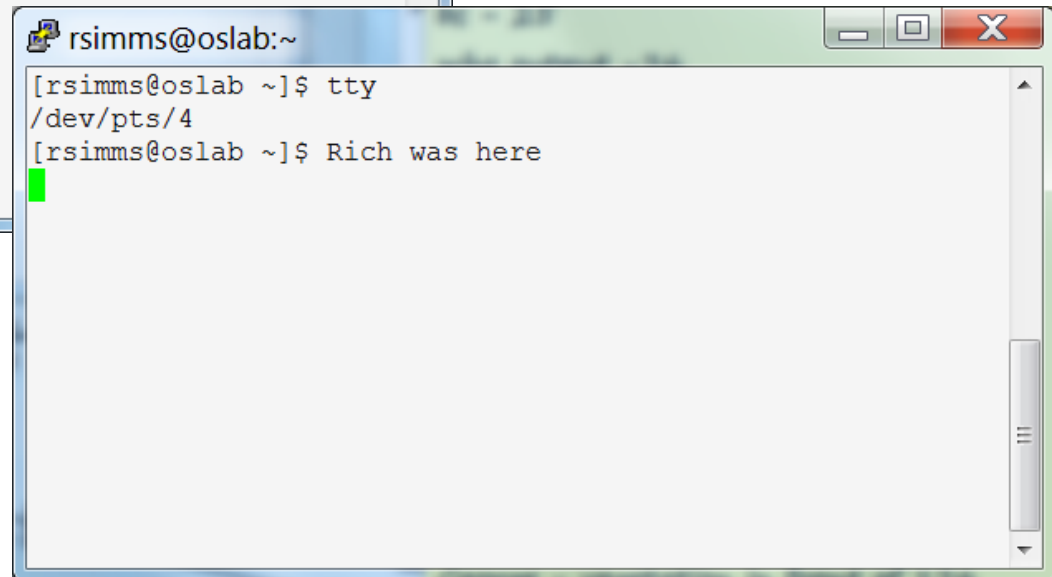


```
rsimms@oslab:~  
[rsimms@oslab ~]$ tty  
/dev/pts/4  
[rsimms@oslab ~]$
```

Everything is a file (even a terminal)

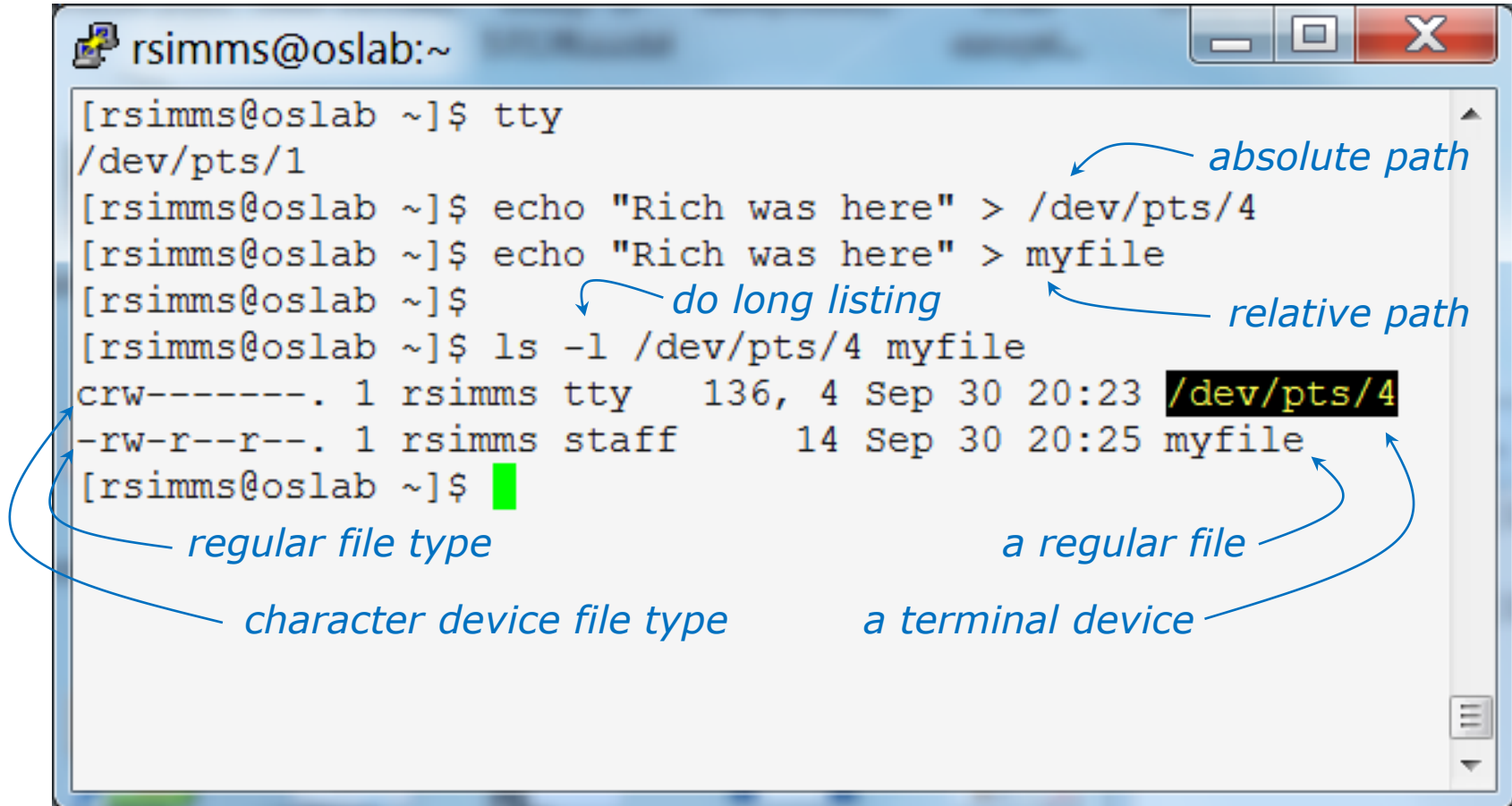


```
rsimms@oslab:~  
[rsimms@oslab ~]$ tty  
/dev/pts/1  
[rsimms@oslab ~]$ echo "Rich was here" > /dev/pts/4  
[rsimms@oslab ~]$
```



```
rsimms@oslab:~  
[rsimms@oslab ~]$ tty  
/dev/pts/4  
[rsimms@oslab ~]$ Rich was here
```

Everything is a file (even a terminal)



```

rsimms@oslab:~
[rsimms@oslab ~]$ tty
/dev/pts/1
[rsimms@oslab ~]$ echo "Rich was here" > /dev/pts/4
[rsimms@oslab ~]$ echo "Rich was here" > myfile
[rsimms@oslab ~]$
[rsimms@oslab ~]$ ls -l /dev/pts/4 myfile
crw-----. 1 rsimms tty    136, 4 Sep 30 20:23 /dev/pts/4
-rw-r--r--. 1 rsimms staff   14 Sep 30 20:25 myfile
[rsimms@oslab ~]$
  
```

absolute path (points to `/dev/pts/4`)

relative path (points to `myfile`)

do long listing (points to `ls -l`)

regular file type (points to the permissions `-rw-r--r--` of `myfile`)

character device file type (points to the permissions `crw-----` of `/dev/pts/4`)

a regular file (points to `myfile`)

a terminal device (points to `/dev/pts/4`)

Class Activity

Part I

- Login into Opus-II
- Use **echo "I can do it" > myfile**
- Print your new file with **cat myfile**

```
simben90@oslab:~
/home/cis90/simben $ echo "I can do it" > myfile
/home/cis90/simben $ cat myfile
I can do it
/home/cis90/simben $ tty
/dev/pts/5
/home/cis90/simben $ cat myfile > /dev/pts/9
/home/cis90/simben $
```

Part II

- Open a second session on Opus-II
- You should have two terminals now
- In both sessions use **tty** to identify the terminal devices being used
- In one terminal use **cat myfile > /dev/pts/xx** where xx is your other terminal

```
simben90@oslab:~
/home/cis90/simben $ tty
/dev/pts/9
/home/cis90/simben $ I can do it
/home/cis90/simben $
```

Did it work?

Click green "Yes" check or red "No" X
on ConferZoom



yes



no



More File Name Expansion

(also known as *globbing*)

Filename Expansion Metacharacters

matches all non-hidden filenames in the current directory when used alone or zero or more characters when used as a prefix, infix or postfix.

?

matches any single character in any of your current directory's filenames.

[]

matches any single character contained within the brackets.



You may also hear this process called "globbing"

Shell Parse Step

Filename expansion happens during the shell parsing step, before the command is even located or executed.

- 1) Prompt
- 2) Parse**
- 3) Search for program (along the path)
- 4) Execute program
- 5) Nap (wait till process is done)
- 6) Repeat

*The commands never see *, ?, and [] expansion characters. These characters get replaced by the shell before the command is even located and executed.*

The * Filename Expansion Metacharacter

Example: **text.*** will be expanded by the shell to match any files that start with "text."

```
/home/cis90/simben $ echo text.*  
text.err text.fxd
```

```
/home/cis90/simben $ ls -i text.*  
19496 text.err 19497 text.fxd
```

```
/home/cis90/simben $ file text.*  
text.err: ASCII text  
text.fxd: ASCII text
```

```
/home/cis90/simben $ wc -l /home/cis90/mahtab/text.*  
11 /home/cis90/mahtab/text.err  
10 /home/cis90/mahtab/text.fxd  
21 total
```

```
/home/cis90/simben $ tail -n1 ../mahtab/text.*  
==> ../mahtab/text.err <==  
number10.
```

```
==> ../mahtab/text.fxd <==  
This is line number 10.
```

The * Filename Expansion Metacharacter

Example: * is expanded to match all directories in /home/cis90 and **ti*** to match all files starting with "ti"

```
/home/cis90/simben $ ls -l ../*/Poems/Blake/ti*
-rw-r--r--. 1 beakie90 cis90 115 Jul 20 2001 ../beakie/Poems/Blake/tiger
-rw-r--r--. 1 calmic90 cis90 115 Jul 20 2001 ../calmic/Poems/Blake/tiger
-rw-r--r--. 1 casenr90 cis90 115 Jul 20 2001 ../casenr/Poems/Blake/tiger
-rw-r--r--. 1 casric90 cis90 115 Jul 20 2001 ../casric/Poems/Blake/tiger
-rw-r--r--. 1 cis90      cis90 115 Jul 20 2001 ../cis/Poems/Blake/tiger
-rw-r--r--. 1 daweli90 cis90 115 Jul 20 2001 ../daweli/Poems/Blake/tiger
-rw-r--r--. 1 fahmic90 cis90 115 Jul 20 2001 ../fahmic/Poems/Blake/tiger
-rw-r--r--. 1 fitcon90 cis90 115 Jul 20 2001 ../fitcon/Poems/Blake/tiger
< snipped >
-rw-r--r--. 1 simben90 cis90 115 Jul 20 2001 ../simben/Poems/Blake/tiger
-rw-r--r--. 1 specod90 cis90 115 Jul 20 2001 ../specod/Poems/Blake/tiger
-rw-r--r--. 1 thinic90 cis90 115 Jul 20 2001 ../thinic/Poems/Blake/tiger
-rw-r--r--. 1 tilbuz90 cis90 115 Jul 20 2001 ../tilbuz/Poems/Blake/tiger
-rw-r--r--. 1 vasjor90 cis90 115 Jul 20 2001 ../vasjor/Poems/Blake/tiger
-rw-r--r--. 1 vivrut90 cis90 115 Jul 20 2001 ../vivrut/Poems/Blake/tiger
-rw-r--r--. 1 weljon90 cis90 115 Jul 20 2001 ../weljon/Poems/Blake/tiger
-rw-r--r--. 1 weltim90 cis90 115 Jul 20 2001 ../weltim/Poems/Blake/tiger
/home/cis90/simben $
```

The * Filename Expansion Metacharacter

Note, DOS uses *.* to match all files.

BUT, this is NOT true in UNIX

```
/home/cis90/simmsben $ echo *.*  
Lab2.0 Lab2.1 text.err text.fxd
```

*Instead, *.* is expanded to match all files in the current directory containing a "."*

The * Filename Expansion Metacharacter

*Note the * metacharacter by itself does not match any hidden files in your current working directory*

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

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


The ? Filename Expansion Metacharacter

Example: **???** will match any three character file name

```
/home/cis90/simben $ echo ???  
bin log
```

```
/home/cis90/simben $ ls bin/???  
bin/app
```

```
/home/cis90/simben $ ls /bin/???  
/bin/awk /bin/csh /bin/env /bin/pwd /bin/red /bin/rvi /bin/tar  
/bin/cat /bin/cut /bin/ksh /bin/raw /bin/rpm /bin/sed
```

```
/home/cis90/simben $ tail -n3 /home/cis90/grodav/bin/???  
while :  
do sleep 1  
done
```

```
/home/cis90/simben $ file /home/cis90/grodav/Poems/*/???  
/home/cis90/grodav/Poems/Yeats/old: ASCII English text
```

The [] Filename Expansion Metacharacter

Example: **[12]** will match a "1" or a "2"

```
/home/cis90/simben $ head -n1 Poems/Shakespeare/sonnet[12]
```

```
==> Poems/Shakespeare/sonnet1 <==
```

```
From fairest creatures we desire increase,
```

```
==> Poems/Shakespeare/sonnet2 <==
```

```
When forty winters shall besiege thy brow,
```

```
/home/cis90/simben $ wc ../balcor/Poems/Shakespeare/sonnet[12]
```

```
14 105 614 ../balcor/Poems/Shakespeare/sonnet1
```

```
14 114 631 ../balcor/Poems/Shakespeare/sonnet2
```

```
28 219 1245 total
```

```
/home/cis90/simben $ ls -d /etc/*[12]*
```

```
/etc/dbus-1 /etc/iproute2 /etc/pnm2ppa.conf /etc/rc2.d
```

```
/etc/DIR_COLORS.256color /etc/mke2fs.conf /etc/polkit-1 /etc/sasl2
```

```
/etc/gtk-2.0 /etc/pbm2ppa.conf /etc/rc1.d /etc/X11
```



File Name Expansion (also known as globbing)

PRACTICE QUESTIONS

The * Filename Expansion Metacharacter

Your turn now

What command would classify all files in the parent directory that start with m?

Write your answer in the chat window

The * Filename Expansion Metacharacter

Answer

What command would classify all files in the parent directory that start with m?

```
/home/cis90/simben $ file ../m*  
../milhom: directory  
../monele: directory  
/home/cis90/simben $
```

Note, the matches can change each term since it is based on student names.

The ? Filename Expansion Metacharacter



A ? matches exactly one character which could be anything

What command would list all 25 character filenames in /bin?

Write your answer in the chat window

The ? Filename Expansion Metacharacter

What command would list all 25 character filenames in /bin?

Answer

```
/home/cis90/simben $ ls /bin/????????????????????????????  
/bin/mysql_secure_installation  /bin/vmware-guestproxycerttool
```

The [] Filename Expansion Metacharacter



A [] will match any character between the brackets

From your home directory, what command would print the first line of all Shakespeare sonnets ending in a 2 or 5?

Write your answer in the chat window

The [] Filename Expansion Metacharacter

From your home directory, what command would print the first line of all Shakespeare sonnets ending in a 2 or 5?

Answer

```
/home/cis90/simben $ head -n 1 Poems/Shakespeare/*[25]
```

```
==> Poems/Shakespeare/sonnet15 <==
```

```
When I consider every thing that grows
```

```
==> Poems/Shakespeare/sonnet2 <==
```

```
When forty winters shall besiege thy brow,
```

```
==> Poems/Shakespeare/sonnet35 <==
```

```
Whoever hath her wish, thou hast thy Will,
```

```
==> Poems/Shakespeare/sonnet5 <==
```

```
Those hours that with gentle work did frame
```

```
/home/cis90/simben $
```

Filename Expansion Metacharacters

* ? []

What commands are there in /usr/bin that start with a "n" or "m", are 5 letters long and end with a "p"?

Hint: Use a combination of filename expansion metacharacters

Write your answer in the chat window

Filename Expansion Metacharacters

* ? []

What commands are there in /usr/bin that start with a "n" or "m", are 5 letters long and end with a "p"?

Answer

```
/home/cis90/simben $ echo /usr/bin/[nm]???p  
/usr/bin/nohup
```

Filename Expansion Metacharacters

* ? []

For the command:

```
file /usr/share/man/*/ [ap]?? .8.gz
```

What arguments does the shell pass to the **file** command to process?

Filename Expansion Metacharacters

* ? []

For the command:

```
file /usr/share/man/*/ [ap]?? .8.gz
```

What arguments does the shell pass to the **file** command to process?

```
/home/cis90/simben $ echo /usr/share/man/*/ [ap]?? .8.gz  
/usr/share/man/man8/arp.8.gz /usr/share/man/man8/atd.8.gz  
/usr/share/man/man8/pam.8.gz /usr/share/man/man8/pvs.8.gz
```

Tip: Use echo to expand complicated filenames containing multiple filename expansion characters

Command Review

Lesson 1 commands:

cal	- show calendar
cat /etc/issue	- usually shows distro (distribution) name
cat /etc/*-release	- usually shows distro (distribution) name
clear	- clear the terminal screen
date	- show current time and date
exit	- terminate your shell and log off
history	- show previous commands
hostname	- show the name of the computer being accessed
id	- show user and group id information
ps	- show processes (loaded programs) being run
ssh	- secure login to a remote system
uname	- show kernel name
tty	- show terminal device
who	- show everyone logged in
who am i	- identifies which login session you are using

*Use the **man** command or google for the details*

Lesson 2 commands:

apropos	- search for string in whatis database
bc	- binary calculator
cat	- print file(s)
cd	- change directory
echo	- print text
env	- show shell environment variables
info	- online documentation with hot links
file	- show file information
ls	- show directory contents
passwd	- change password
set	- show (or set) shell variables
type	- show command location in path
man	- manual page for a command
whatis	- command summary

New Files and Directories:

/etc/passwd	- user accounts
/etc/shadow	- encrypted passwords
/bin	- directory of commands
/sbin	- directory of superuser commands
/usr/bin	- directory of commands, tools and utilities
/usr/sbin	- directory of superuser commands, tools and utilities
/usr/local/bin	- custom local commands

Lesson 3 commands:

mail

<integer>
[Enter key]
p <message list>
d <message list>
s <message list> file
u <message list>
R <message list>
r <message list>
m <user list>
q
x
h
z or z-

mesg

write

irssi

- UNIX mail

print specific message specified by <integer>
print next message
print messages
delete messages
save (append) messages to file
undelete messages
reply to sender
reply to all
mail to specific users
quit saving changes
exit without saving changes
print message headers
scroll forward or backward through headers

- Enable or disable writes to your terminal
- Write message to another user
- IRC chat client

New Files and Directories:

/var/mail

/var/mail/*username*

mbox

- Message store for mail
- Incoming mailbox for *username*
- File in users home directory where read messages are archived to

Lesson 4 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:

/	"slash" directory, the root of the file tree
/home	User home directories
/home/cis90	CIS 90 class home directories
/home/cis90/ <i>username</i>	The home directory for CIS 90 student <i>username</i>
/etc/passwd	The absolute pathname of the passwd file in the /etc/ directory

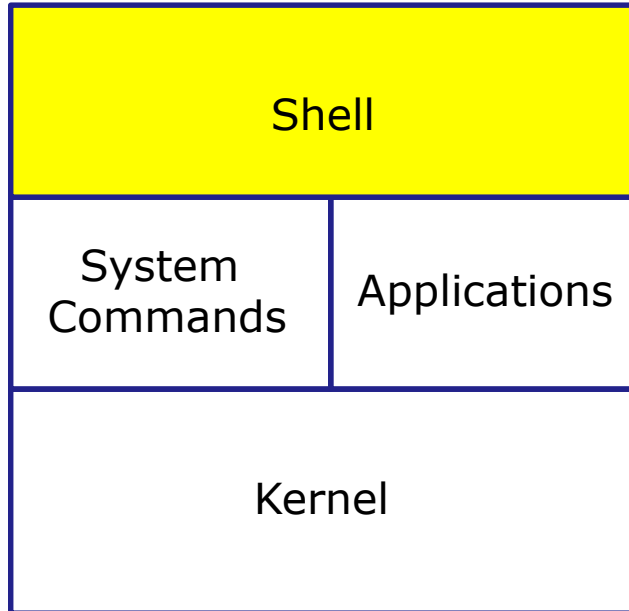


Command line Syntax & Parsing

(review)



Life of the Shell



- 1) **Prompt** for a command
- 2) **Parse** (interpret metacharacters, expand file names and dissect command line into options, arguments and redirection)
- 3) **Search** for program (along the path)
- 4) **Execute** program by loading into memory (becomes a process), hookup input and outputs, and pass along command line options and arguments.
- 5) **Nap** (wait till process is done)
- 6) **Repeat**

Command Syntax

Command**Options****Arguments****Redirection**

Command – is the name of an executable program file.

Options – various options which control how the program will operate.

Arguments – the objects the command is directed to work upon.

Redirection – The default input stream (stdin) is from the console keyboard, the default output (stdout) and error (stderr) streams go to the console screen. Redirection can modify these streams to other files or devices.

Command Syntax

*Shell prints
this to prompt
user to enter a
command*

Shell parses this command line



Examples

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

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

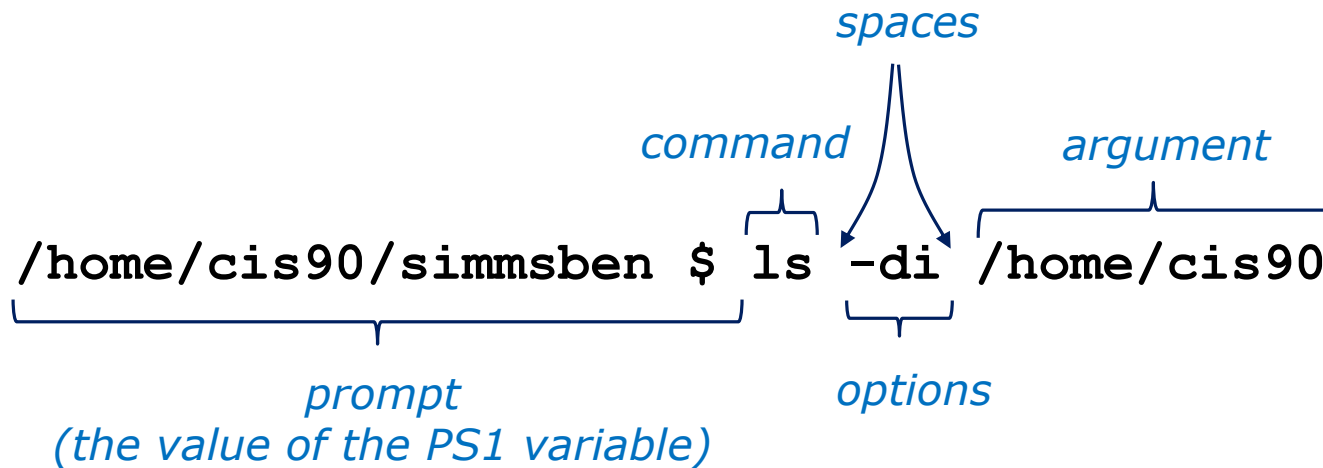
***Redirection** is covered
later in the course*

```

/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.*

Command Line Syntax Review



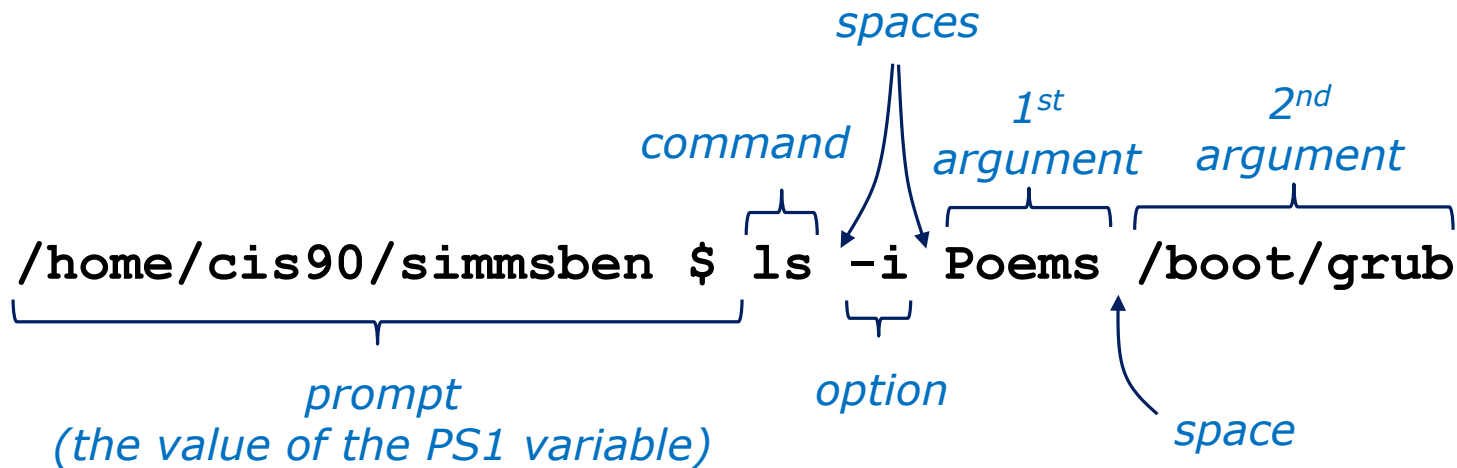
Parsing the command line above yields:

One command: **ls**

Two options: **d** and **i**

One argument: **/home/cis90** (an absolute pathname to a directory)

Command Line Syntax Review



Parsing the command line above yields:

One command: **ls**

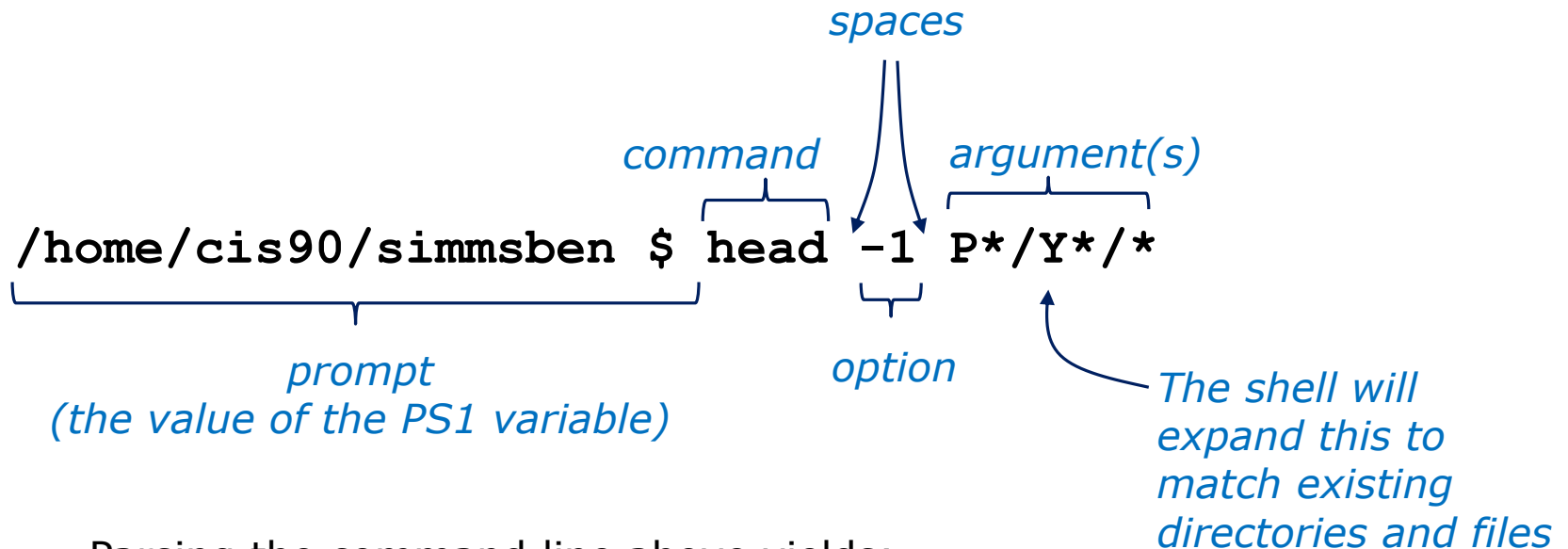
One options: **i**

Two arguments:

Poems (a relative pathname to a directory)

/boot/grub (an absolute pathname to a directory)

Command Line Syntax Review



Parsing the command line above yields:

One command: **head**

One option: **1**

Three arguments:

Poems/Yeats/mooncat (a relative pathname to a file)

Poems/Yeats/old (a relative pathname to a file)

Poems/Yeats/whitebirds (a relative pathname to a file)



Command line Syntax & Parsing

PRACTICE QUESTIONS

Your turn now!

```
head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
```

Parse the command above and identify and see if you can identify:

1. The command
2. The options
3. The number of arguments:
4. The actual arguments:

Your turn now!

```
head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
```

Parse the command above and identify:

1. The command: **head**

The command is the head command

Your turn now!

```
head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
```

Parse the command above and identify:

1. The command: **head**
2. The options: **-n1**

The options are -n1 which the user added to instruct the head command to output only one line

Your turn now!

```
head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
```

Parse the command above and identify:

1. The command: **head**
2. The options: **-n1**
3. The number of arguments: 5

Note, the number of arguments can change each term since it is based on student names. How many are there today?

The number of arguments is not apparent without first processing all the filename expansion characters. You can use the echo command to see exactly how the expansion will be done.

```
/home/cis90/simben $ echo /home/cis90/s??*/P*/Shake*/s????t[36]  
/home/cis90/samwil/Poems/Shakespeare/sonnet3 /home/cis90/schmic/Poems/Shakespeare/sonnet3  
/home/cis90/seasky/Poems/Shakespeare/sonnet3 /home/cis90/siljas/Poems/Shakespeare/sonnet3  
/home/cis90/simben/Poems/Shakespeare/sonnet3
```

Your turn now!

```
head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
```

Parse the command above and identify:

1. The command: **head**
2. The options: **-n1**
3. The number of arguments: 5
4. The actual arguments:

```
/home/cis90/samwil/Poems/Shakespeare/sonnet3  
/home/cis90/schmic/Poems/Shakespeare/sonnet3  
/home/cis90/seasky/Poems/Shakespeare/sonnet3  
/home/cis90/siljas/Poems/Shakespeare/sonnet3  
/home/cis90/simben/Poems/Shakespeare/sonnet3
```

Output from the command

```
/home/cis90/simben $ head -n1 /home/cis90/s??*/P*/Shake*/s????t[36]
==> /home/cis90/samwil/Poems/Shakespeare/sonnet3 <==
Look in thy glass and tell the face thou viewest,

==> /home/cis90/schmic/Poems/Shakespeare/sonnet3 <==
Look in thy glass and tell the face thou viewest,

==> /home/cis90/seasky/Poems/Shakespeare/sonnet3 <==
Look in thy glass and tell the face thou viewest,

==> /home/cis90/siljas/Poems/Shakespeare/sonnet3 <==
Look in thy glass and tell the face thou viewest,

==> /home/cis90/simben/Poems/Shakespeare/sonnet3 <==
Look in thy glass and tell the face thou viewest,
/home/cis90/simben $
```

Note: the shell expanded /home/cis90/s??/P*/Shake*/s????t[36] into five arguments which match the sonnet 3 files belonging to all students whose last name starts with a "s".*



Meta Characters (review)

Metacharacters

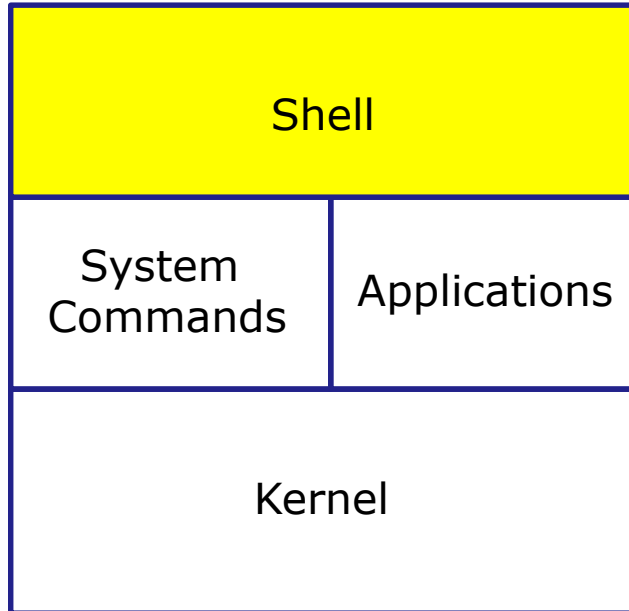
Have special interpretation by the shell

Char	Description
\	Treat the following metacharacter as a plain character. Also called "escaping" the next character.
\$	The following text is a shell (environment) variable and the value should be used.
<cr>	Carriage return marks the end of the command
;	Separates multiple commands on one line
'	used to enclose a string that the shell will not do further interpretation
"	Used to enclose a string that the shell will do further interpretation.
>	Redirects stdout (more in Lesson 8)
2>	Redirects stderr (more in Lesson 8)
*	Matches all non-hidden file names when used alone or zero or more characters when used as prefix, infix or postfix
?	Matches any single character of a file name
[]	Matches any single character contained within the brackets
#	Not an official metacharacter, but any text following the # is ignored by the shell



Life of the Shell

*The shell processes metacharacters during the **Parse** step*



- 1) **Prompt** for a command
- 2) **Parse** (interpret metacharacters, expand file names and dissect command line into options, arguments and redirection)
- 3) **Search** for program (along the path)
- 4) **Execute** program by loading into memory (becomes a process), hookup input and outputs, and pass along command line options and arguments.
- 5) **Nap** (wait till process is done)
- 6) **Repeat**

Metacharacters

#

has the ability to make everything that follows the # be ignored by the shell. Good for adding comments in scripts

```
/home/cis90/simmsben $ #OK lets escape the carriage return in next example  
/home/cis90/simmsben $
```

Note there is no error message because everything after the # is ignored

Metacharacters

\$

\$ metacharacter has the ability to "show the value of"

```
/home/cis90/simmsben $ EYES=brown  
/home/cis90/simmsben $ echo EYES  
EYES  
/home/cis90/simmsben $ echo $EYES  
brown  
  
/home/cis90/simmsben $ echo $LOGNAME  
simmsben  
/home/cis90/simmsben $
```

echo the string EYES

echo the value of the variable EYES

echo the value of the predefined environment variable LOGNAME

Metacharacters

" and '

Weak "double" quotes allow the shell to process \$ metacharacters inside the quoted string

```
/home/cis90/simmsben $ echo "I am in $PWD"  
I am in /home/cis90/simmsben
```

```
/home/cis90/simmsben $ echo 'I am in $PWD'  
I am in $PWD  
/home/cis90/simmsben $
```

Strong "single" quotes block the shell from processing \$ metacharacters inside the quoted string

Metacharacters

;

```
/home/cis90/simmsben $ #Lets put two commands on one line  
/home/cis90/simmsben $ echo "This is my terminal device:"; tty  
This is my terminal device:  
/dev/pts/2  
/home/cis90/simmsben $
```

the ; metachacter lets you combine several commands on one line

Metacharacters

\

```
/home/cis90/simmsben $ #OK lets escape the carriage return in next example  
/home/cis90/simmsben $ echo Lets start line 1 here \  
> and finish it here  
Lets start line 1 here and finish it here  
/home/cis90/simmsben $
```



*The \ is used to escape the next character typed.
Use an escape to disable the special abilities of a metacharacter.*

Escaping a carriage return (the Enter key) tells the shell to keep inputting more characters from the next line for the current command being entered.

Metacharacters

\

Escaping the # means it is no longer treated as comment

```
/home/cis90/simmsben $ \#OK lets put a comment here
-bash: #OK: command not found
/home/cis90/simmsben $
/home/cis90/simmsben $
/home/cis90/simmsben $ echo $PS1
$PWD $
/home/cis90/simmsben $ echo \$PS1
$PS1
/home/cis90/simmsben $
```

and you get an error when the shell processes your comment

Escaping the \$ means \$ is no longer treated "the value of"

Environment Variables

(review)

Shell (Environment) Variables

common environment variables

Shell Variable	Description
HOME	Users home directory (starts here after logging in and returns with a <code>cd</code> command (with no arguments)
LOGNAME	User's username for logging in with.
PATH	List of directories, separated by <code>:</code> 's, for the Shell to search for commands (which are program files) .
PS1	The prompt string.
PWD	Current working directory
SHELL	Name of the Shell program being used.
TERM	Type of terminal device , e.g. dumb, vt100, xterm, ansi, etc.

Shell (Environment) Variables

Show variable values

```
/home/cis90/simben $ echo $HOME  
/home/cis90/simben
```

```
/home/cis90/simben $ echo $LOGNAME  
simben90
```

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

*Use echo to show the
values of variables*

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

```
/home/cis90/simben $ echo $SHELL  
/bin/bash
```

```
/home/cis90/simben $ echo $TERM  
xterm
```

Shell (Environment) Variables

PATH

```
/home/cis90/simben $ echo $PATH
```

```
/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/  
cis90/simben/../../bin:/home/cis90/simben/bin:.
```

These are the directories in Benji's PATH in the order they will be searched:

1st: /usr/local/bin
2nd: /bin
3rd: /usr/bin
4th: /usr/local/sbin
5th: /usr/sbin
6th: /home/cis90/simben/../../bin
7th: /home/cis90/simben/bin
8th: .

The PATH variable is used by the shell to locate commands

Shell (Environment) Variables

Set variable values

Use an "=" with no spaces to set values of variables

```
/home/cis90/simben $ # Change the prompt variable
/home/cis90/simben $ PS1='[\u@\h \W]\$ '
[simben90@opus-ii ~]$ echo $PS1
[\u@\h \W]\$
[simben90@opus-ii ~]$
```

```
[simben90@opus-ii ~]$ # Change it back again
[simben90@opus-ii ~]$ PS1='$PWD $ '
/home/cis90/simben $ echo $PS1
$PWD $
/home/cis90/simben $
```

Shell Variables

Set variable values

If the variable has never been used before then it is created

```
/home/cis90/simben $ myfavoritedog="Benji"  
/home/cis90/simben $ echo $myfavoritedog  
Benji
```

Shell (Environment) Variables

env command – show all environment variables

```
/home/cis90/simmsben/Poems $ env
HOSTNAME=opus-ii.cabrillo.edu
SHELL=/bin/bash
TERM=xterm
HISTSIZE=1000
USER=simmsben
LS_COLORS=no=00:fi=00:di=00;34:ln=00;36:pi=40;33:so=00;35:bd=40;33;01:cd=40;33;01:or=01;05;37;41:mi=01;05;37;41:ex=00;32:*.cmd=00;32:*.exe=00;32:*.com=00;32:*.btm=00;32:*.bat=00;32:*.sh=00;32:*.csh=00;32:*.tar=00;31:*.tgz=00;31:*.arj=00;31:*.taz=00;31:*.lzh=00;31:*.zip=00;31:*.z=00;31:*.Z=00;31:*.gz=00;31:*.bz2=00;31:*.bz=00;31:*.tz=00;31:*.rpm=00;31:*.cpio=00;31:*.jpg=00;35:*.gif=00;35:*.bmp=00;35:*.xbm=00;35:*.xpm=00;35:*.png=00;35:*.tif=00;35:
USERNAME=
MAIL=/var/spool/mail/simmsben
PATH=/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/home/cis90/simmsben/../../bin:/home/cis90/simmsben/bin:
INPUTRC=/etc/inputrc
PWD=/home/cis90/simmsben/Poems
LANG=en_US.UTF-8
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
SHLVL=1
HOME=/home/cis90/simmsben
BASH_ENV=/home/cis90/simmsben/.bashrc
LOGNAME=simmsben
CVS_RSH=ssh
LESSOPEN=|/usr/bin/lesspipe.sh %s
G_BROKEN_FILENAMES=1
_=/bin/env
OLDPWD=/home/cis90/simmsben
/home/cis90/simmsben/Poems $
```

*Use the **env** command to show all environment variables (a subset of the shell variables)*

Shell Variables

set command – show all shell variables

/home/cis90/simmsben/Poems \$ **set | more**

```
BASH=/bin/bash
BASH_ARGC=()
BASH_ARGV=()
BASH_ENV=/home/cis90/simmsben/.bashrc
BASH_LINENO=()
BASH_SOURCE=()
BASH_VERSINFO=([0]="3" [1]="2" [2]="25" [3]="1" [4]="release"
[5]="i686-redhat-linux-gnu")
BASH_VERSION='3.2.25(1)-release'
COLORS=/etc/DIR_COLORS.xterm
COLUMNS=80
CVS_RSH=ssh
DIRSTACK=()
EUID=1160
GROUPS=()
G_BROKEN_FILENAMES=1
HISTFILE=/home/cis90/simmsben/.bash_history
HISTFILESIZE=1000
HISTSZ=1000
HOME=/home/cis90/simmsben
HOSTNAME=opus-ii.cabrillo.edu
HOSTTYPE=i686
IFS=$' \t\n'
IGNOREEOF=10
INPUTRC=/etc/inputrc
LANG=en_US.UTF-8
LESSOPEN='|/usr/bin/lesspipe.sh %s'
LINES=24
LOGNAME=simmsben
```

```
LS_COLORS='no=00:fi=00:di=00;34:ln=00;36:pi=40;33:so=00;35
:bd=40;33;01:cd=40;33;01:or=01;05;37;41:mi=01;05;37;41:ex=
00;32:*.cmd=00;32:*.exe=00;32:*.com=00;32:*.btm=00;32:*.ba
t=00;32:*.sh=00;32:*.csh=00;32:*.tar=00;31:*.tgz=00;31:*.a
rj=00;31:*.taz=00;31:*.lzh=00;31:*.zip=00;31:*.z=00;31:*.Z
=00;31:*.gz=00;31:*.bz2=00;31:*.bz=00;31:*.tz=00;31:*.rpm=
00;31:*.cpio=00;31:*.jpg=00;35:*.gif=00;35:*.bmp=00;35:*.x
bm=00;35:*.xpm=00;35:*.png=00;35:*.tif=00;35:'
MACHTYPE=i686-redhat-linux-gnu
MAIL=/var/spool/mail/simmsben
MAILCHECK=60
OLDPWD=/home/cis90/simmsben
OPTERR=1
OPTIND=1
OSTYPE=linux-gnu
PATH=/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/home/
cis90/simmsben/./bin:/home/cis90/simmsben/bin:.
PIPESTATUS=([0]="0")
PPID=26514
PROMPT_COMMAND='echo -ne
"\033]0;${USER}@${HOSTNAME}%.*}:${PWD/#$HOME/~}"; echo -ne
"\007"'
PS1='$PWD $'
PS2='> '
PS4='+ '
PWD=/home/cis90/simmsben/Poems
SHELL=/bin/bash
SHELLOPTS=braceexpand:emacs:hashall:histexpand:ignoreeof:i
nteractive-comments:monitor
SHLVL=1
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
TERM=xterm
UID=1160
USER=simmsben
USERNAME=
_=_env
consoletype=pty
<snipped>
```

*Use the **set** command to show all shell variables (which includes the environment variables)*

Class Exercise

- Change your prompt with:
PS1='\$LOGNAME, command please: '
- Change your prompt with:
PS1='[\u@\h \W]\\$ '
- Change your prompt with:
PS1="\$PWD \$ "
Now change directories using **cd**, what happened?
- Restore original prompt with:
PS1='\$PWD \$ '

Did it work?

Click green "Yes" check or red "No" X
on ConferZoom



yes



no



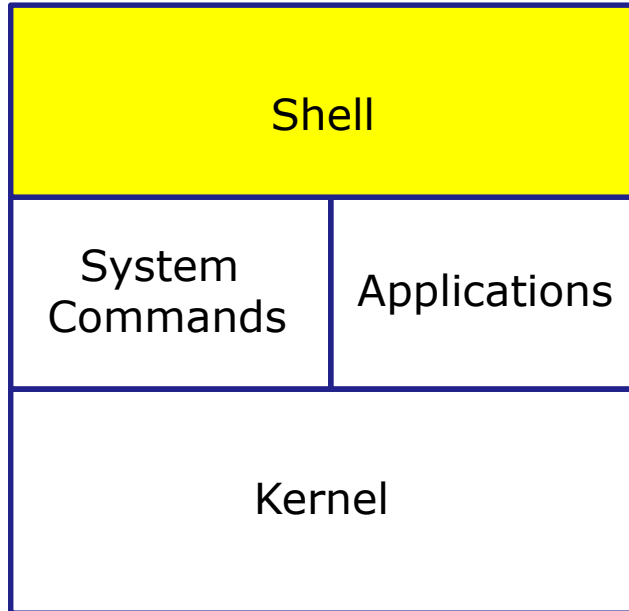
Inputs & Outputs

(continuing)



Life of the Shell

Execution begins only if the command is found



- 1) **Prompt** for a command
- 2) **Parse** (interpret metacharacters, expand file names and dissect command line into options and arguments)
- 3) **Search** for program (along the path)
- 4) **Execute** program by loading into memory (becomes a process), hookup input and outputs, and pass along command line options and arguments.
- 5) **Nap** (wait till process is done)
- 6) **Repeat**



Step 4 - the shell executes the command program file

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

*The next step is to load the **file** command that was found into memory. The program on the hard drive becomes a **process** in memory with a unique PID (Process ID).*

*Each new process is given three file descriptors **stdin**, **stdout** and **stderr** for input and output purposes.*

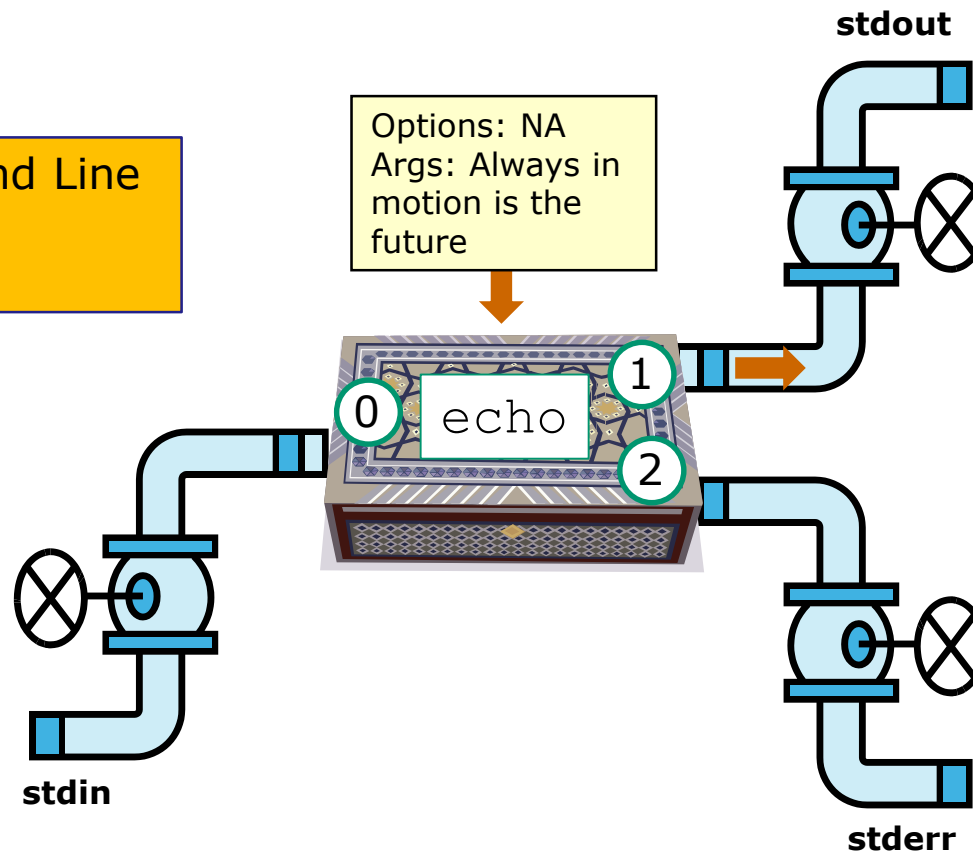
*These are sometimes referred to as the three **standard IO (Input/Output) streams**.*

Example program to process: echo command

```
[rsimms@opus-ii ~]$ echo Always in motion is the
future
Always in motion is the future
[rsimms@opus-ii ~]$
```

Inputs: Command Line

Outputs: stdout



`/dev/pts/1`



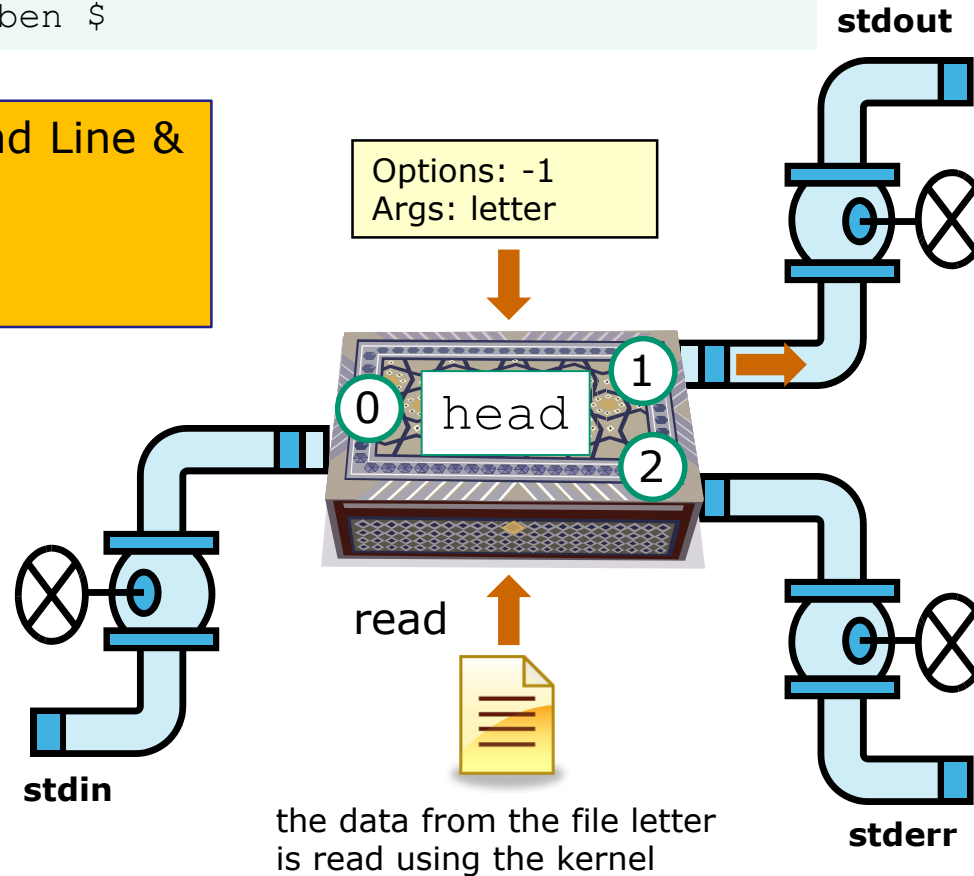
Always in
motion is
the future

Example program to process: head command

```
/home/cis90/simmsben $ head -1 letter
Hello Mother! Hello Father!
/home/cis90/simmsben $
```

Inputs: Command Line & Operating System

Outputs: stdout



/dev/pts/1



Hello Mother!
Hello Father!

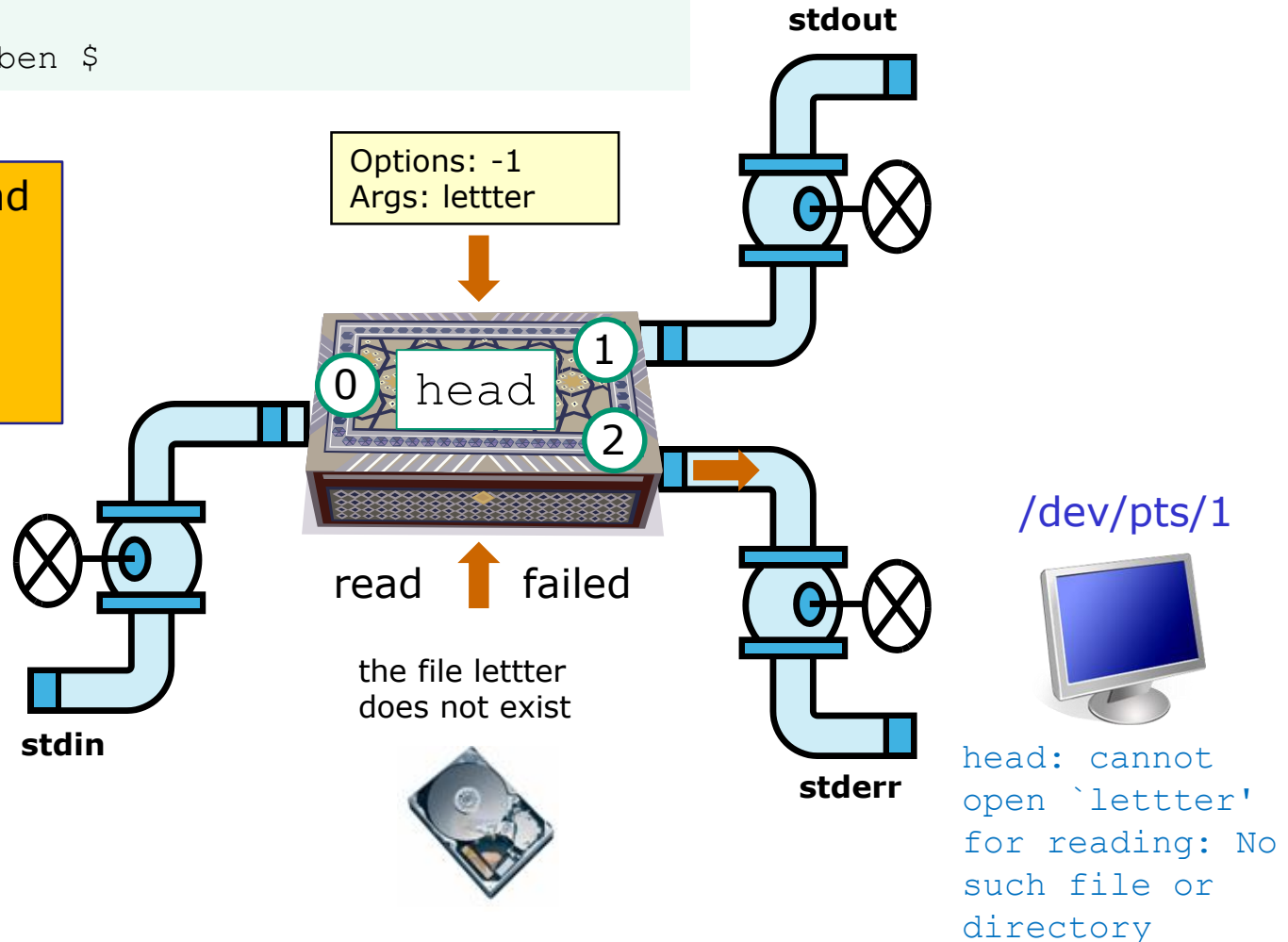


Example program to process: head command

```
/home/cis90/simmsben $ head -1 lettter
head: cannot open `lettter' for reading: No such
file or directory
/home/cis90/simmsben $
```

Inputs: Command
Line & Operating
System

Outputs: stderr

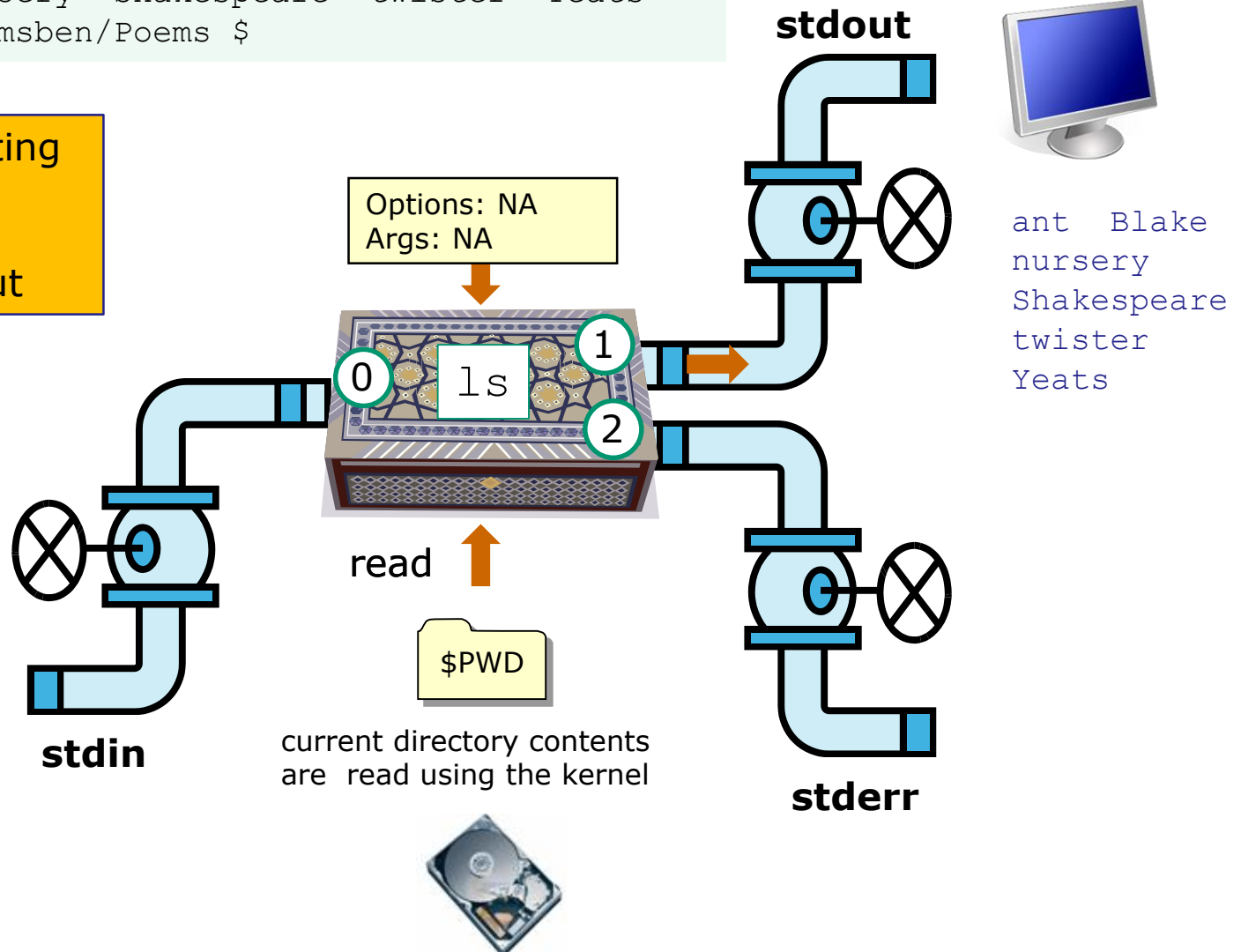


Example program to process: ls command

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

Inputs: Operating System

Outputs: stdout



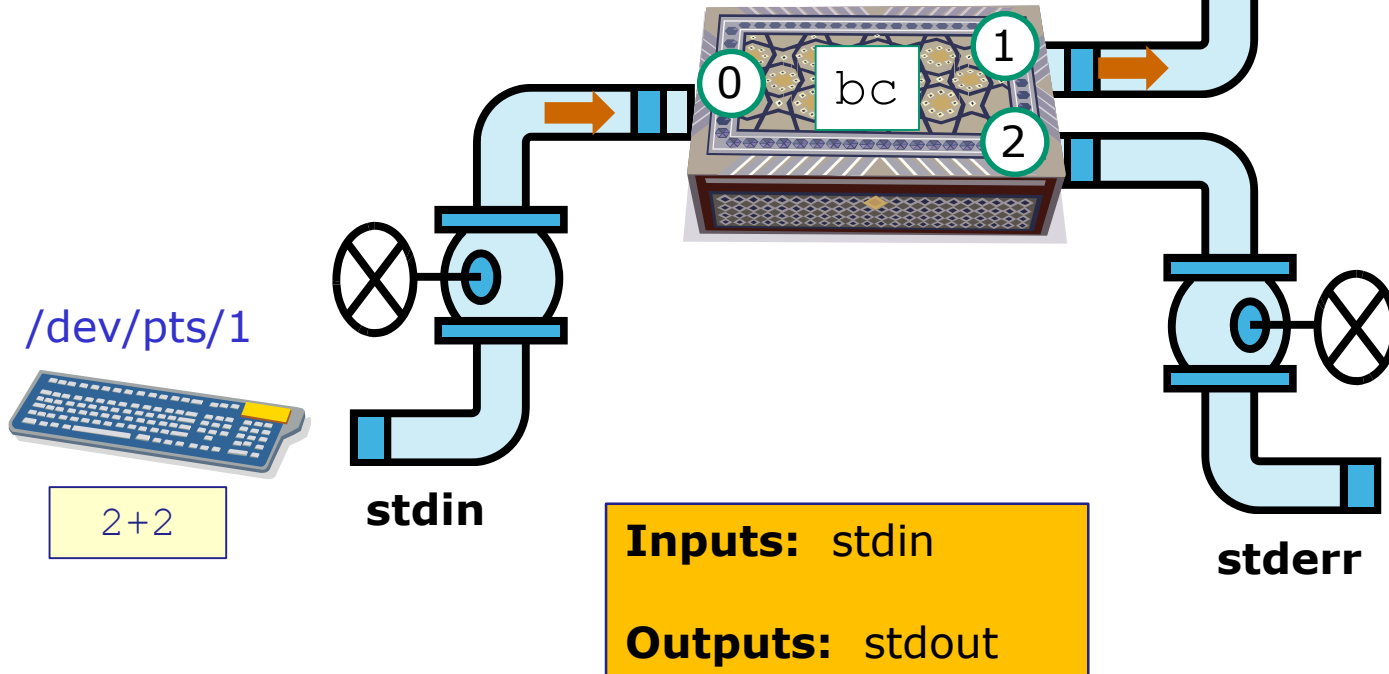
Example program to process: bc command

```
/home/cis90/simmsben $ bc
bc 1.06
Copyright 1991-1994, 1997, 1998, 2000 Free Software
Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
2+2
4
```

/dev/pts/1



```
bc 1.06
Copyright 1991-
1994, 1997,
1998, 2000 Free
Software
Foundation, Inc.
This is free
software with
ABSOLUTELY NO
WARRANTY.
For details type
`warranty'.
4
```

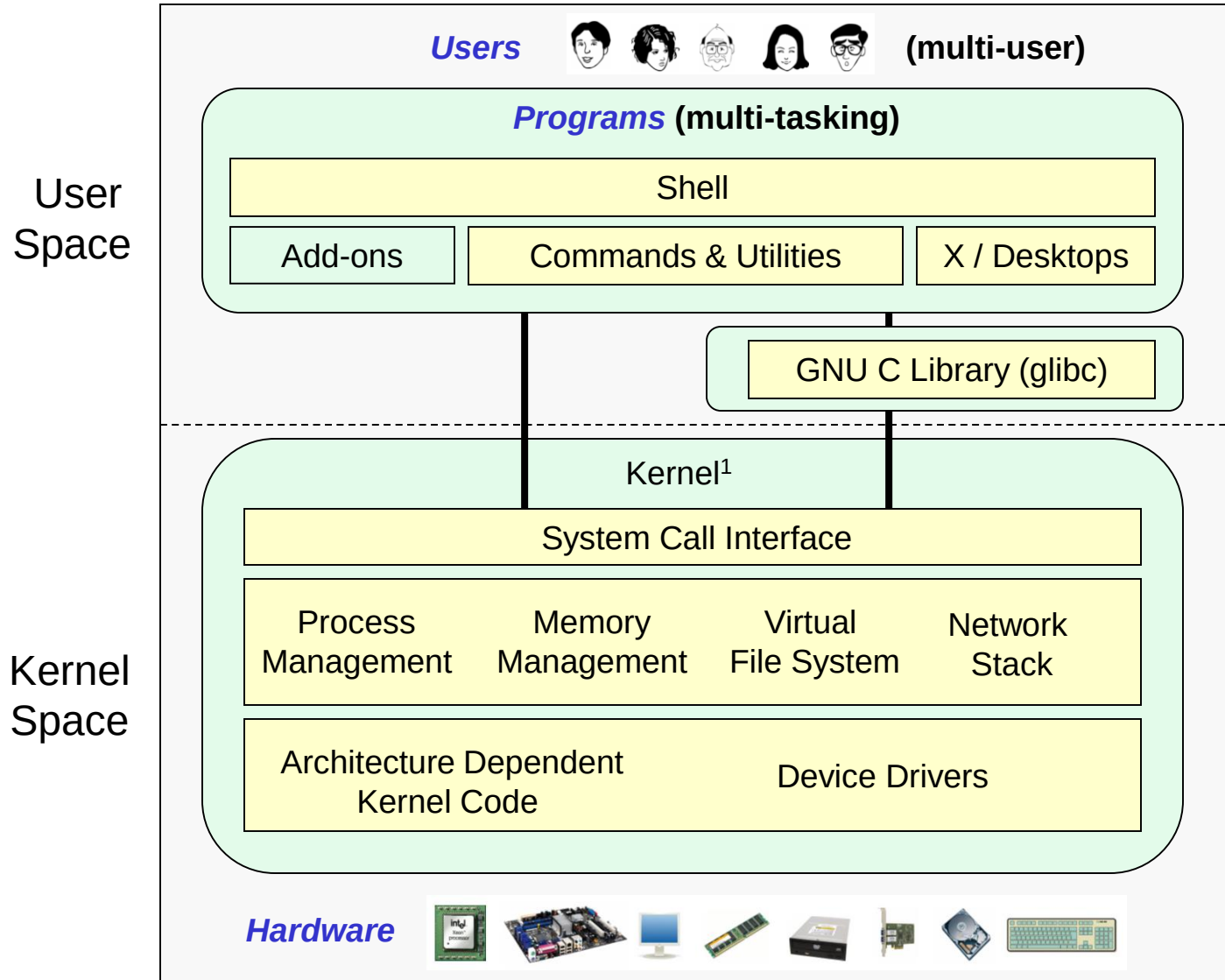




The Kernel



GNU/Linux Operating System Architecture



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

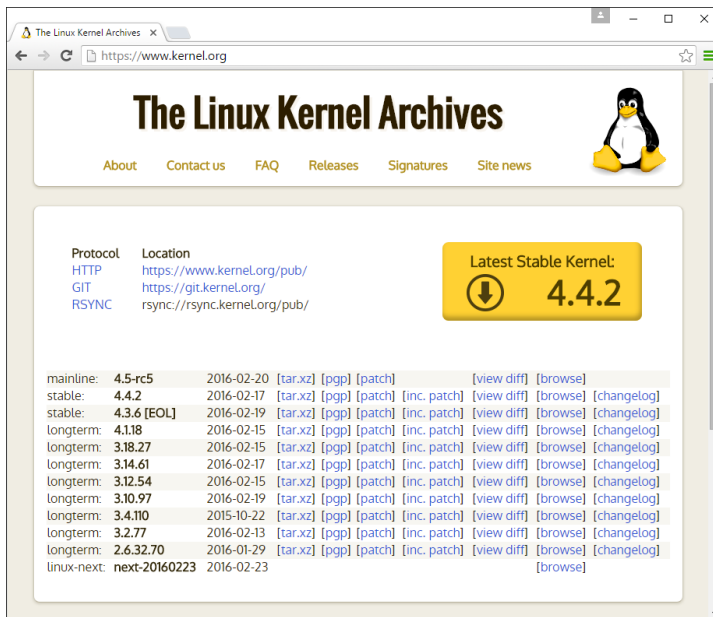


Linus Torvalds, as a student, initially conceived and assembled the Linux kernel in 1991. The kernel was later re-licensed under the GNU General Public License in 1992.

¹See "Anatomy of the Linux kernel" by M. Tim Jones at <http://www-128.ibm.com/developerworks/linux/library/l-linux-kernel/>

The Source for Linux Kernels

*Use the **-r** option on **uname** to see which release of the kernel is running on your system*



The first three numbers x.y.z indicate which kernel is being used from kernel.org.

x is the kernel version
y is the major revision number
z is the minor revision number

Anything after the dash was added by the distribution to indicate a modified kernel used by that distribution.

<https://www.kernel.org/>

```
/home/cis90/simben $ uname -r    cis90@Arya-02:~$ uname -r
3.10.0-514.26.2.el7.x86_64        3.13.0-44-generic
```

The 3.13.0 kernel on Arya is newer than the older 3.10.0 kernel on Opus-II

The kernel files are stored in the /boot directory

```
/home/cis90/simben $ uname -r
```

```
3.10.0-514.26.2.el7.x86_64
```

```
/home/cis90/simben $ ls /boot/vmlinuz*
```

```
/boot/vmlinuz-0-rescue-2d84cd08bb7441d7b41ae5ff5cba84df /boot/vmlinuz-3.10.0-514.26.2.el7.x86_64  
/boot/vmlinuz-3.10.0-514.26.1.el7.x86_64 /boot/vmlinuz-3.10.0-693.2.2.el7.x86_64
```

```
/home/cis90/simben $ file /boot/vmlinuz-3.10.0-514.26.2.el7.x86_64
```

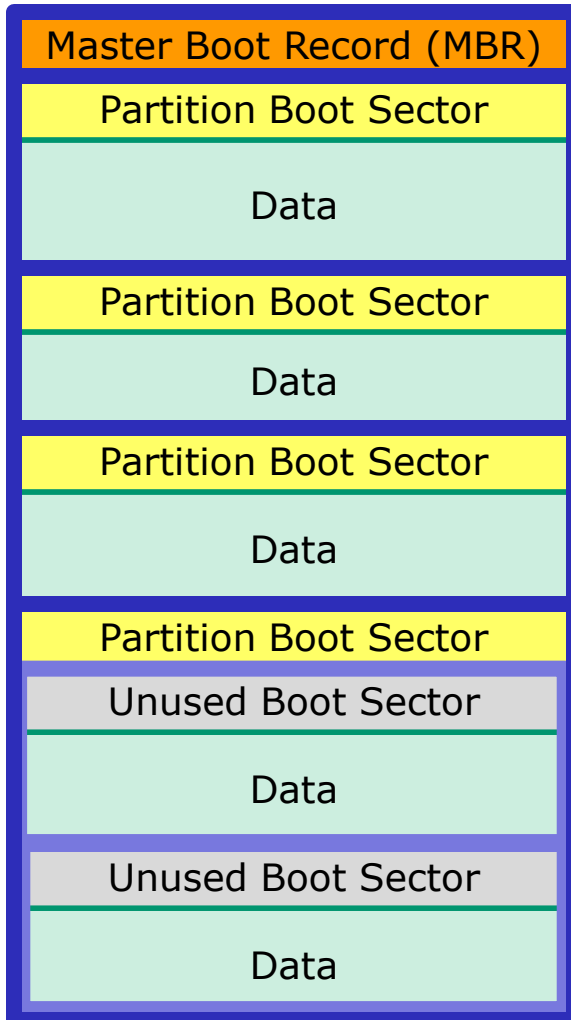
```
/boot/vmlinuz-3.10.0-514.26.2.el7.x86_64: Linux kernel x86 boot executable  
bzImage, version 3.10.0-514.26.2.el7.x86_64 (builder@kbuilder.dev.centos.org)  
#1, RO-rootFS, swap_dev 0x5, Normal VGA
```

*The **uname -r** command indicates the release of the kernel that is running. This will correspond to one of the kernel files in the boot directory.*

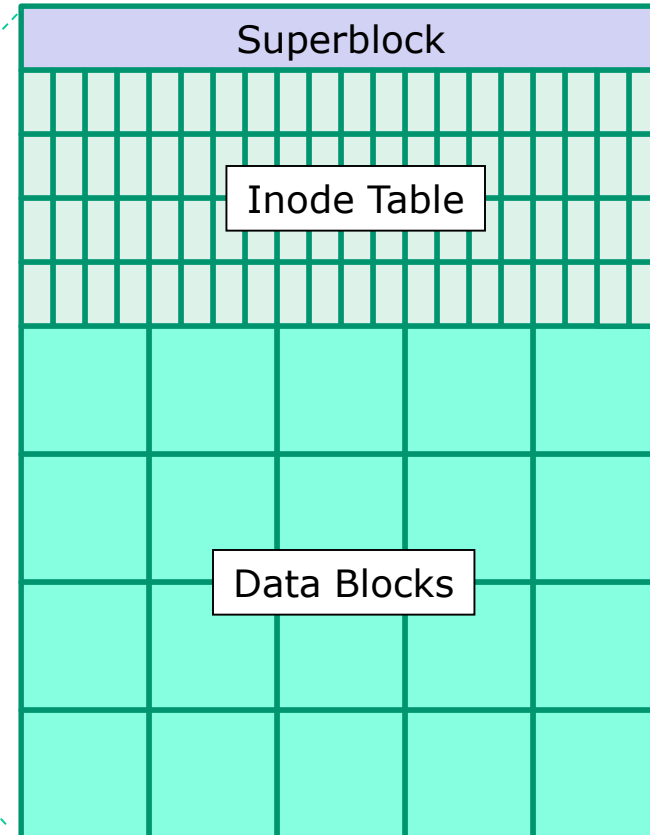
File System (review)

File Systems

Linux



ext2 file system



The three elements of a UNIX 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?"
```

filename

+

inode

+

data

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

bigfile 19470
bin 9628
letter 9662

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.

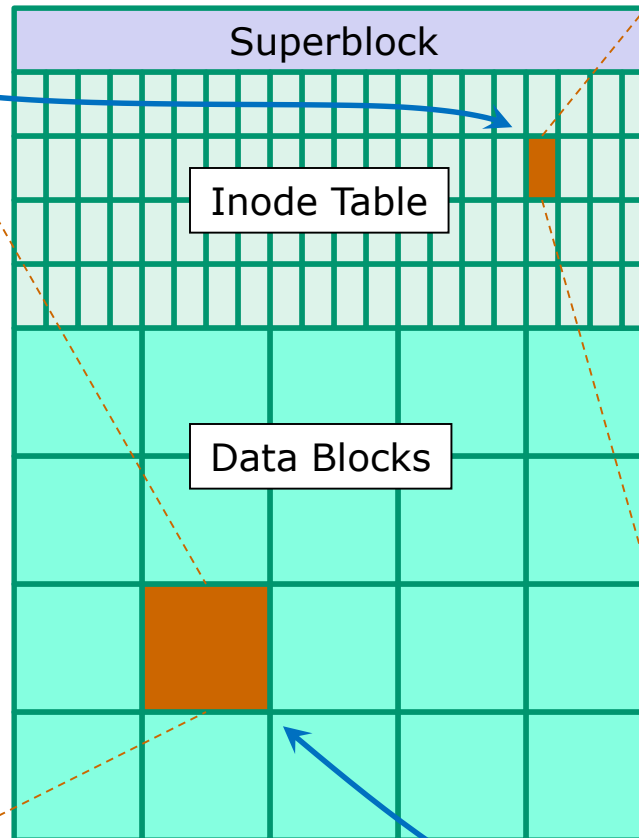
Take me home, oh Mother, Father, take me home! I hate Granada.
Don't leave me out in the forest where I might get eaten by a bear! Take me home, I promise that I won't make noise, or mess the house with other boys, oh please don't make me stay -- I've been here one whole day.

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

ext2 file system



9662	inode number
-	Type
rw-r--r--	Permissions
1	Number of links
simben90	User
cis90	Group
1044	Size
2001-07-20	Modification time
2012-09-17	Access Time
2012-08-01	Change time
Pointer(s) to data blocks	Pointer(s) to data blocks

```
/home/cis90/simmsben $ ls -il letter
```

```
9662 -rw-r--r--. 1 simben90 cis90 1044 Jul 20 2001 letter
```

Basic File Types and Commands

Long listing code (ls -l)	Type	How to make one
d	directory	mkdir
-	regular • Programs • Text • Data (binary) <i>Use the file command to further classify files</i>	touch
l	symbolic link	ln -s
c	character device files	mknod
b	block device files	mknod

Note: Other files types includes sockets (s) and named pipes (p)

Interpreting a long listing file types

The terminal window shows the output of the `ls -la` command in the directory `/home/cis90/simmsben`. The output lists various files and directories with their permissions, owner, group, size, date, and name. Annotations on the right side of the terminal explain the meaning of the first character in the permission string (column 1) and the color of the file name.

File Type	File Name	Permissions	Owner	Group	Size	Date	Time
Directory	.	drwx-----	simmsben	cis90	4096	Aug 8	11:51
Directory	..	drwxr-x---	rsimms	cis90	4096	Jun 30	14:57
Regular File	.bash_history	-rw-----	simmsben	cis90	11409	Aug 7	19:20
Regular File	.bash_logout	-rw-----	simmsben	cis90	24	Jul 20	2001
Regular File	.bash_profile	-rw-----	simmsben	cis90	354	Sep 17	2003
Regular File	.bashrc	-rw-----	simmsben	cis90	146	Jan 18	2004
Regular File	bcommands	-rw-rw-r--	simmsben	cis90	56	Jul 8	17:22
Regular File	bigfile	-rw-r--r--	simmsben	cis90	10576	Jul 20	2001
Regular File	bin	drwxr-xr-x	simmsben	cis90	4096	Sep 11	2005
Regular File	deleteme	-rw-rw-r--	simmsben	cis90	1044	Aug 8	11:52
Regular File	.emacs	-rw-r--r--	simmsben	cis90	515	Jun 30	14:57
Regular File	empty	-rw-r--r--	simmsben	cis90	0	Jul 20	2001
Directory	Hidden	d-----	simmsben	cis90	4096	Feb 1	2002
Regular File	Lab2.0	drwxr-xr-x	simmsben	cis90	4096	Feb 17	2001
Regular File	Lab2.1	drwxr-xr-x	simmsben	cis90	4096	Feb 17	2001
Regular File	.lessht	-rw-----	simmsben	cis90	35	Aug 8	13:58
Regular File	letter	-rw-r--r--	simmsben	cis90	1044	Jul 20	2001
Regular File	mbox	-rw-----	simmsben	cis90	5799	Jul 24	21:08
Regular File	Miscellaneous	drwxr-xr-x	simmsben	cis90	4096	Sep 11	2005
Regular File	mission	-rw-r--r--	simmsben	cis90	759	Jun 6	2002
Regular File	.mozilla	drwxr-xr-x	simmsben	cis90	4096	Jun 30	14:57
Regular File	.plan	-rw-r--r--	simmsben	cis90	40	Jul 20	2001
Regular File	Poems	drwxr-xr-x	simmsben	cis90	4096	Jul 9	14:24
Regular File	proposal1	-rw-r--r--	simmsben	cis90	1074	Aug 26	2003
Regular File	proposal2	-rw-r--r--	simmsben	cis90	2175	Jul 20	2001
Regular File	proposal3	-rw-r--r--	simmsben	cis90	2054	Sep 14	2003
Regular File	results-el	-rw-r--r--	simmsben	cis90	5467	Jul 6	13:41
Regular File	results-el.a	-rw-r--r--	simmsben	cis90	1286	Jul 6	12:20
Regular File	salsa	-rw-rw-r--	simmsben	cis90	688	Jul 24	15:35
Regular File	small_town	-rw-r--r--	simmsben	cis90	1580	Nov 16	2004
Regular File	spellk	-rw-r--r--	simmsben	cis90	485	Aug 26	2003
Regular File	text.err	-rw-r--r--	simmsben	cis90	250	Jul 20	2001
Regular File	text.fxd	-rw-r--r--	simmsben	cis90	231	Jul 20	2001
Regular File	timecal	-rwxr-xr-x	simmsben	cis90	509	Jun 6	2002
Regular File	.viminfo	-rw-----	simmsben	cis90	661	Jul 24	13:59
Regular File	what_am_i	-rw-r--r--	simmsben	cis90	352	Jul 20	2001
Regular File	.Xauthority	-rw-----	simmsben	cis90	126	Aug 7	14:23
Regular File	.zshrc	-rw-r--r--	simmsben	cis90	658	Jun 30	14:57

Annotations:

- All directories in the UNIX file tree contain these two hidden . and .. directories (d in column 1)
- A regular file (- in column 1) Its hidden because it starts with a .
- A directory (d in column 1) Color is blue because it's a directory
- A hidden directory (d in column 1, name starts with .)
- Regular file (- in column 1)
- regular file (- in column 1) Color is green because with execute bits are set

Use the **file** command to get additional information about a file

Symbolic links

*A symbolic link file
(l in column 1)*

```
/home/cis90/simben $ ls -l accounts /etc/passwd
lrwxrwxrwx 1 simben90 cis90 11 Mar 7 08:52 accounts -> /etc/passwd
-rw-r--r-- 1 root root 7183 Mar 6 08:17 /etc/passwd
/home/cis90/simben $
```

```
/home/cis90/simben $ head -5 /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

The accounts file in Benji's directory is a symbolic link to the /etc/passwd file.

```
/home/cis90/simben $ head -5 accounts
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

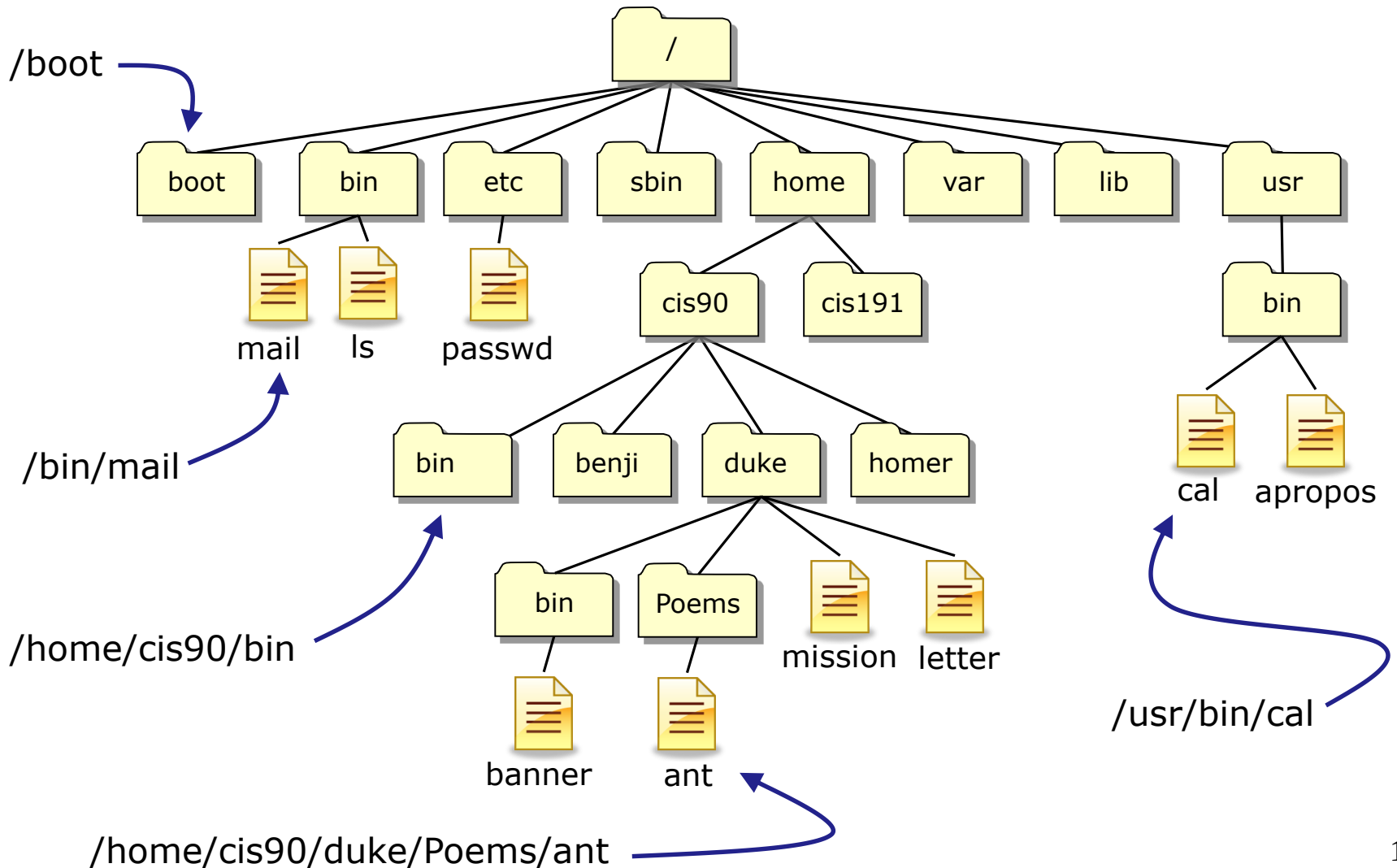
These "shortcuts" can be used for convenience

```
/home/cis90/simben $ ls -li accounts /etc/passwd
99983 accounts 1280173 /etc/passwd
/home/cis90/simben $
```

Note they have different inodes

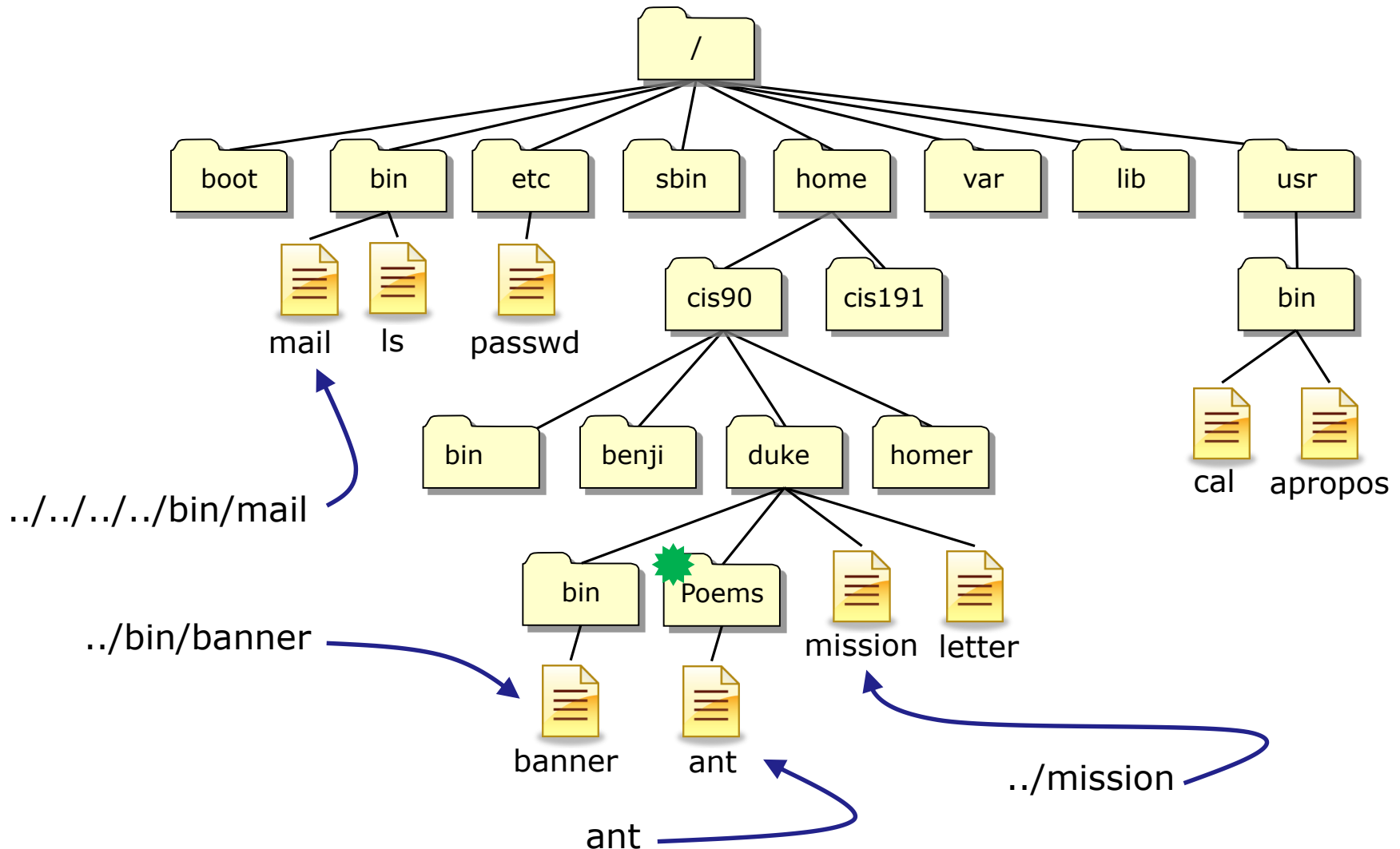
Absolute Pathnames

Start with from /



Relative Pathnames

Start from your current location in the tree



Top Level Directories

Directory	Contents
/bin	binary files forming the commands and shells used by the system administrator and users
/boot	files used during the initial boot-up process including the kernel
/dev	device files 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

Absolute Pathname Target Practice



Analyze the absolute pathname

What directory is the file in?

What is the name of the file in that directory?

Type your answers in the chat window



Make Teams

CCC Confer Breakout Rooms Test



Everyone needs to be on ConferZoom today

Instructor team maker: `/home/rsimms/scripts/teams`

The screenshot shows a virtual breakout room interface. On the left, a sidebar contains a video feed of 'Rich Simms', a 'PARTICIPANTS' list with 'Benji' and 'ROOM 1' (checked), and a 'CHAT' window. The main area displays a 'redhat Room 2' logo, a terminal window with the command `echo $LOGNAME` outputting `simben90`, and three images: a roasted chicken, a stack of cookies, and a tray of colorful cookies. Annotations with arrows point to specific elements: 'Use this to put a green check next to your name' points to the checkmark in the participants list; 'Use the camera icon to do screen captures' points to the camera icon in the sidebar; 'Use normal copy and paste as well from your desktop' points to the terminal and the images.

Use this to put a green check next to your name

Use the camera icon to do screen captures

Use normal copy and paste as well from your desktop

I'll be sending you into virtual breakout rooms today so you can work together on various activities

Flashcards

Lessons L1-L5 random



Flashcards
Deck size " "
L1-L5
All categories
L1=18
L2=22
L3=5
L4=26
L5=4
Total=75

Rules

- Chat window belongs to team that is up (no one else can use)
- "Final Answer" must be from someone on team that hasn't answered yet
- All team members can help each other and suggest answers

```
Instructor timer: /home/rsimms/scripts/countdown  
i=15; while [ $i -gt 0 ]; do clear; banner $i; let i=i-1; sleep 1; done; clear; banner done
```

Flash Cards

Click on Flashcards in left panel

Rich's Cabrillo College CIS Classes Login Page

Home Resources Forums CIS Lab CTC

Please Login
You need to login first

Username:
Password:

Login

New users click [here](#)

Metal Sitemap W3C XHTML 1.0 W3C CSS Credits Earth

Register if this is the first time using Flashcards

Rich's Cabrillo College CIS Classes Registration

Home Resources Forums CIS Lab CTC

Registration

First Name:
Last Name:
Email:

Create your login credentials

Username:
Password:
Password again:

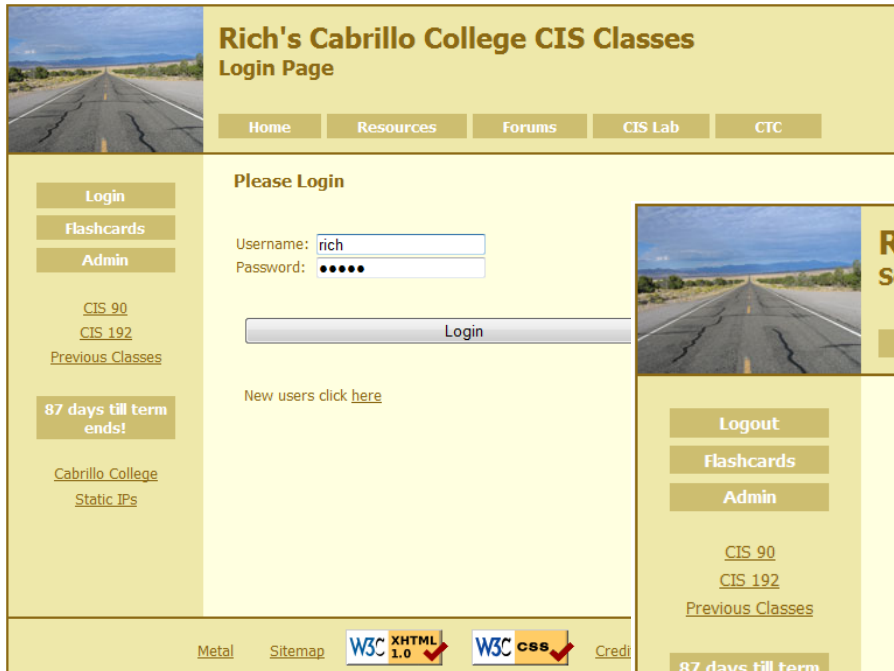
Submit

Metal Sitemap W3C XHTML 1.0 W3C CSS Credits Earth

Register and choose a username and password of your choice

Logging in and using Flashcards

Login with your username and password



Rich's Cabrillo College CIS Classes
Login Page

Home Resources Forums CIS Lab CTC

Please Login

Username:
Password:

Login

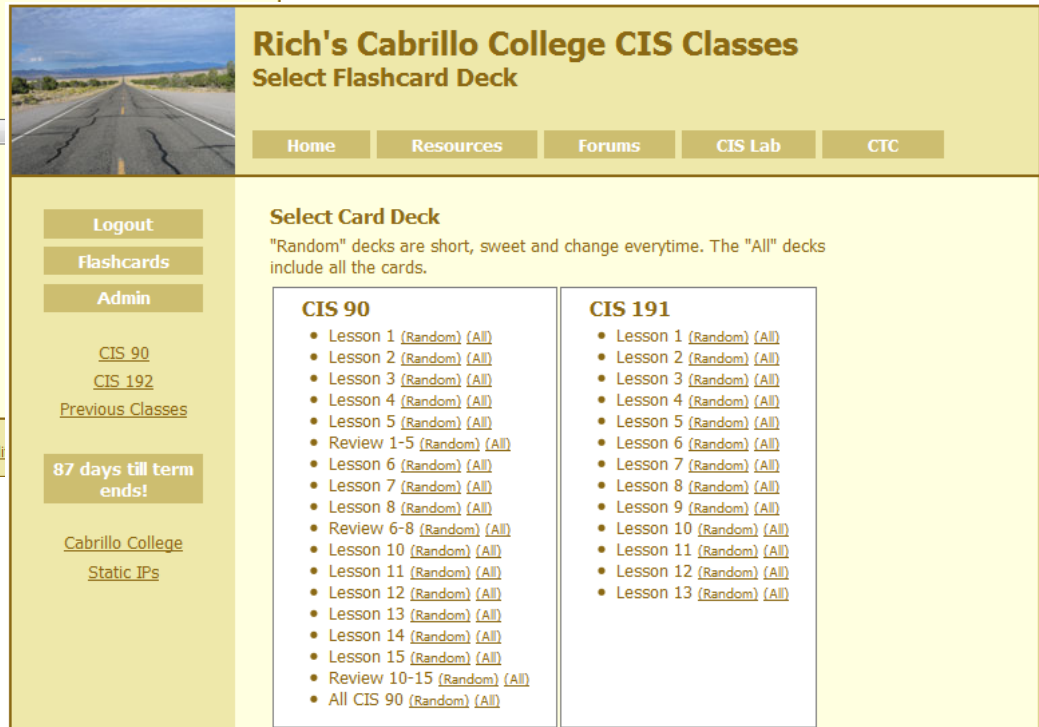
New users click [here](#)

87 days till term ends!

[Cabrillo College](#)
[Static IPs](#)

Metal Sitemap W3C XHTML 1.0 W3C CSS Credit

Select deck of cards



Rich's Cabrillo College CIS Classes
Select Flashcard Deck

Home Resources Forums CIS Lab CTC

Select Card Deck

"Random" decks are short, sweet and change everytime. The "All" decks include all the cards.

CIS 90	CIS 191
• Lesson 1 (Random) (All)	• Lesson 1 (Random) (All)
• Lesson 2 (Random) (All)	• Lesson 2 (Random) (All)
• Lesson 3 (Random) (All)	• Lesson 3 (Random) (All)
• Lesson 4 (Random) (All)	• Lesson 4 (Random) (All)
• Lesson 5 (Random) (All)	• Lesson 5 (Random) (All)
• Review 1-5 (Random) (All)	• Lesson 6 (Random) (All)
• Lesson 6 (Random) (All)	• Lesson 7 (Random) (All)
• Lesson 7 (Random) (All)	• Lesson 8 (Random) (All)
• Lesson 8 (Random) (All)	• Lesson 9 (Random) (All)
• Review 6-8 (Random) (All)	• Lesson 10 (Random) (All)
• Lesson 10 (Random) (All)	• Lesson 11 (Random) (All)
• Lesson 11 (Random) (All)	• Lesson 12 (Random) (All)
• Lesson 12 (Random) (All)	• Lesson 13 (Random) (All)
• Lesson 13 (Random) (All)	
• Lesson 14 (Random) (All)	
• Lesson 15 (Random) (All)	
• Review 10-15 (Random) (All)	
• All CIS 90 (Random) (All)	

Class Exercise

Flashcards

- Browse to simms-teach.com
- Register with a username and password of your choice
- Verify you can login and use the flash cards.

Test Tips

What command ... ?

Tips on how to answer questions on lab assignments and tests

What command will do “blah, blah, blah” questions:

Examples:

- What **ls** command allows you to see the permissions of your home directory while you are in your home directory?
- What command will give you a prompt showing your current working directory path and a \$?
- What command allows you to see hidden files in your current directory?

*Tip: Always use Opus-II (or the appropriate VM) to test your answers for these kinds of questions. **I will!** If your command doesn't work it won't be the right answer!*

Practice Question

What **ls** command allows you to see the permissions of your home directory while you are in your home directory?

Practice Question

What **ls** command allows you to see the permissions of your home directory while you are in your home directory?

```
/home/cis90/simben $ ls -l
total 392
-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 606 Feb 29 22:17 dead.letter
-rw-r--r-- 1 simben90 cis90 0 Jul 20 2001 empty
d----- 2 simben90 cis90 4096 Feb 1 2002 Hidden
< snipped >
-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 25390 Feb 29 22:18 uhistory

-rw-r--r-- 1 simben90 cis90 352 Mar 5 08:24 what_am_i
/home/cis90/simben $
```

Nope, that didn't work. We got permissions of all the files in the directory but we didn't get the permissions of the directory itself!

Practice Question

What **ls** command allows you to see the permissions of your home directory while you are in your home directory?

```
/home/cis90/simben $ ls -dl /home/cis90/simben  
drwxr-xr-x 10 simben90 cis90 4096 Mar  1 10:15
```

```
/home/cis90/simben $ ls -dl ~  
drwxr-xr-x 10 simben90 cis90 4096 Mar  1 10:15
```

```
/home/cis90/simben $ ls -dl .  
drwxr-xr-x 10 simben90 cis90 4096 Mar  1 10:15 .
```

```
/home/cis90/simben $ ls -dl $HOME  
drwxr-xr-x 10 simben90 cis90 4096 Mar  1 10:15
```

```
/home/cis90/simben $ ls -dl  
drwxr-xr-x 10 simben90 cis90 4096 Mar  1 10:15 .
```

Yep, they all worked! The *-d* option instructs the *ls* command not to descend into the directory. Any of the commands above would be correct.

Practice Question

What command will give you a prompt showing your current working directory path and a \$?

Practice Question

What command will give you a prompt showing your current working directory path and a \$?

```
/home/cis90/simben $ PS1=blah
blah
blahPS1="/home/cis90/simben $ "
```

```
/home/cis90/simben $
/home/cis90/simben $ cd ..
/home/cis90/simben $ cd
/home/cis90/simben $
/home/cis90/simben $ echo $PS1
/home/cis90/simben $
```

Nope, that didn't work. The prompt doesn't change after changing to another directory

Practice Question

What command will give you a prompt showing your current working directory path and a \$?

```
/home/cis90/simben $ PS1=blah
blah
blahPS1="PWD $ "
PWD $
PWD $ echo $PS1
PWD $
```

Nope, that didn't work either. A \$ in front of the variable name is required to use its value.

Practice Question

What command will give you a prompt showing your current working directory path and a \$?

```
PWD $ PS1=blah
blah
blahPS1="$PWD $ "
/home/cis90/simben $ cd ..
/home/cis90/simben $ cd
/home/cis90/simben $
/home/cis90/simben $ echo $PS1
/home/cis90/simben $
```

Better, but still didn't work. The prompt is still not changing after cd'ing to another directory.

We need to block bash from expanding the \$PWD variable when it's being set.

Practice Question

What command will give you a prompt showing your current working directory path and a \$?

```
/home/cis90/simben $ PS1=blah
blah
blahPS1='$PWD $ '
/home/cis90/simben $ cd ..
/home/cis90 $ cd
/home/cis90/simben $
/home/cis90/simben $ echo $PS1
$PWD $
```

Touchdown! That worked!

The single quotes prevent bash from expanding \$PWD when setting the PS1 variable.

It is not expanded till the prompt is actually generated for the next command.

Practice Question

What command allows you to see hidden files in your current directory?

Practice Question

What command allows you to see hidden files in your current directory?

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

bigfile	lab01.graded	Lab2.1	mission	small_town	uhistory
bin	lab01-submitted	letter	Poems	spellk	what_am_i
dead.letter	lab02.graded	log	proposal1	text.err	
empty	lab03.graded	mbox	proposal2	text.fxd	
Hidden	Lab2.0	Miscellaneous	proposal3	timecal	

Nope, that didn't work! Hidden files start with a "." and note of these start with a "." (period)

Practice Question

What command allows you to see hidden files in your current directory?

```
/home/cis90/simben $ ls -a
```

.	dead.letter	Lab2.0	.mozilla	.ssh
..	.emacs	Lab2.1	.plan	text.err
.bash_history	empty	.lesshst	Poems	text.fxd
.bash_logout	Hidden	letter	proposal1	timecal
.bash_profile	lab01.graded	log	proposal2	uhistory
.bashrc	lab01-submitted	mbox	proposal3	.viminfo
bigfile	lab02.graded	Miscellaneous	small_town	what_am_i
bin	lab03.graded	mission	spellk	

Bingo, that worked! Hidden files and directories start with a "." (period)

How many arguments
or “parse this
command” questions

Tips on how to answer questions on lab assignments and tests

How many arguments or “parse this command” questions

Example: The shell performs file name expansion during the Parse step. When a user types the command: **file /v*/l??/*o*.[14]** on Opus-II, how many arguments get passed to the **file** command? What specifically are those arguments?

Tip: Use the echo command to preview how the shell will expand arguments containing metacharacters.

***Tip:** Use the echo command to preview how the shell will expand arguments containing metacharacters.*

***TIP:** Use the echo command to preview how the shell will expand arguments containing metacharacters.*

***TIP:** Use the echo command to preview how the shell will expand arguments containing metacharacters.*

Practice Question

The shell performs file name expansion during the Parse step. When a user types the command: **file /v*/l??/*o*[14]** on Opus-II, how many arguments get passed to the **file** command? What specifically are those arguments?

```
/home/cis90/simben $ echo /v*/l??/*o*[14]  
/var/lib/polkit-1 /var/log/dracut.log-20130101 /var/log/yum.log-20130101
```

*Answer: The shell will expand **/v*/l??/*o*[14]** into the 3 arguments shown above*

Practice Question

Parse the following command on Opus-II:

```
wc -wl /home/cis90/d*t/*w*
```

what is the second argument passed to the **wc** command?

Practice Question

Parse the following command on Opus-II:

```
wc -wl /home/cis90/d*t/*w*
```

what is the second argument passed to the **wc** command?

command: **wc**

options: **w** and **l**

arguments:

```
[rsimms@oslab ~]$ echo /home/cis90/d*t/*w*
```

```
/home/cis90/depot/network /home/cis90/depot/newfile /home/cis90/depot/randomwords
```

*3 arguments, the
second argument is* 

Answer: /home/cis90/depot/newfile

Practice Question

Parse the following command on Opus-II:

```
wc -wl /home/cis90/d*t/*w*
```

Regarding the options passed to the wc command, how many and what are they?

Practice Question

Parse the following command on Opus-II:

```
wc -wl /home/cis90/d*t/*w*
```

Regarding the options passed to the wc command, how many and what are they?

command: wc

options: w and l

arguments:

/home/cis90/depot/network

/home/cis90/depot/newfile

/home/cis90/depot/randomwords

Answer: there are two options, w and l

Absolute/relative pathname questions:

Tips on how to answer questions on lab assignments and tests

Absolute/relative pathname questions:

Examples:

- What is the relative pathname from your home directory to the **date** command?
- What is the absolute path to the sonnet1 file in your Shakespeare directory?

*Tip: Use the **ls** command with tab completion to verify your absolute or relative pathnames*

*Tip: Use the **ls** command with tab completion to verify your absolute or relative pathnames*

*Tip: Use the **ls** command with tab completion to verify your absolute or relative pathnames*

Tip: Use the **ls** command with tab completion to verify your absolute or relative pathnames

Practice

What is the relative pathname from your home directory to the **date** command?

Practice

What is the relative pathname from your home directory to the **date** command?

First, use the type command to find where the date command is

```
/home/cis90/simmsben $ type date
date is /bin/date
```

```
/home/cis90/simben $ ls ../
ahrmatt/      colabd/      huljef/      olscam/      rodduk/
answers/      deltas/      jimmel/      pacnan/      shidev/
.bash_profile depot/      lowmic/      phacha/      simben/
bin/          doucor/      macrya/      plajos/      varana/
blerav/       flamatt/      maxsco/      plajua/      veleli/
bodian/       gueous/      mcidar/      porjon/
bunsol/       guest/       milhen/      pummas/
cheken/       helrog/      milhom/      rafdav/
cofcol/       hovdav/      milmic/      reedie/

/home/cis90/simben $ ls ../../
backup/      cis191/      cis90/      guest/      rick/      turnin/
cis164/      cis192/      cis98/      jimg/      rsimms/    .Xauthority
cis172/      cis193/      gerlinde/   mikki/      ryan/

/home/cis90/simben $ ls ../../../../
.autofsck  etc/      media/      opt/      selinux/  tmp/
bin/       home/     misc/       proc/     srv/      u/
boot/      lib/      mnt/        root/     sys/      usr/
dev/       lost+found/ net/        sbin/     tftpboot/ var/

/home/cis90/simben $ ls ../../../../bin/date
../../../../bin/date
```

Tap tab key twice to see what is in that directory

No errors so this relative pathname is GOOD!

Answer: ../../../../bin/date

Example

What is the absolute path to the sonnet1 file in your Shakespeare directory?

Practice

What is the absolute path to the sonnet1 file in your Shakespeare directory?

```
/home/cis90/simben $ ls /
.autofsck  etc/      media/    opt/      selinux/  tmp/
bin/       home/     misc/     proc/     srv/      u/
< snipped >
/home/cis90/simben $ ls /home/
backup/    cis191/    cis90/    guest/    rick/     turnin/
< snipped >
/home/cis90/simben $ ls /home/cis90/
ahrmat/    colabd/    huljef/    olscam/    rodduk/
answers/    deltas/    jimmel/    pacnan/    shidev/
.bash_profile depot/    lowmic/    phacha/    simben/
< snipped >
cofcol/    hovdav/    milmic/    reedie/
/home/cis90/simben $ ls /home/cis90/simben/
.bash_history lab01.graded Miscellaneous/ .ssh/
< snipped >
.bashrc      lab03.graded .plan      timecal
bigfile      Lab2.0/      Poems/     uhistory
< snipped >
Hidden/      mbox      spellk
/home/cis90/simben $ ls /home/cis90/simben/Poems/
ant          Blake/    nursery    Shakespeare/ twister    Yeats/
/home/cis90/simben $ ls /home/cis90/simben/Poems/Shakespeare/sonnet
sonnet1  sonnet11  sonnet17  sonnet26  sonnet35  sonnet5  sonnet9
sonnet10 sonnet15  sonnet2   sonnet3   sonnet4   sonnet7
/home/cis90/simben $ ls /home/cis90/simben/Poems/Shakespeare/sonnet1
/home/cis90/simben/Poems/Shakespeare/sonnet1
```

*Tap tab key
twice to see
what is in that
directory*

No errors so this absolute pathname is GOOD!

Assignment





How to prepare for the test:

- Review slides for Lessons 1-5 (download and make sure you know how to electronically search PDFs)
- **DO THE PRACTICE TEST**
- Compare your practice test answers with others and discuss on the forum
- **DO THE PRACTICE TEST AGAIN**
- Note the steps you take to answer each question so you can use them again on the real test
- **DO THE PRACTICE TEST AGAIN**
- Go through the Lesson 1-5 flashcards till you feel comfortable with the material
- **DO THE PRACTICE TEST AGAIN**
- Practice, practice, practice ... repeating Labs 1-4 never hurts!

The practice test systems will be shutdown shortly before the real test starts. You can take the practice test and many times as you want until then. Try to get your time down to no more than 30 seconds per question.



P = 5 minutes before class ends

Splashdown = 30 minutes before real test starts

Reminder to instructor:

On Canvas

- Schedule Practice Test from P to Splashdown
- Publish Practice Test
- Remove password on practice test
- Update Q16, Q22 and Q30 as needed

On Practice Test primary system

- `echo "/root/unlock-cis90; rm /etc/nologin" | at P`

On Practice Test secondary systems

- `echo /root/unlock-cis90 | at P (adjust for timezones)`

On Opus-II

- `echo "/home/rsimms/cis90/test01/q29/mail-q29-P1 2 q" | at P+5`

Practice Test

A practice test is available on Canvas.

The screenshot shows the Canvas LMS interface. On the left is a blue sidebar with navigation links: Home, Account, Dashboard, Courses, Groups, Calendar, and Inbox. The main content area is titled 'CIS-90 (Room 828) > Quizzes'. It features a search bar and a section for 'Assignment Quizzes'. A table lists the available quizzes:

Quiz Name	Available until	Due	Points	Questions
CIS 90 Test 1 (practice)	Available until Mar 2	Due Mar 2 at 10:30am	33 pts	33 Questions

A blue arrow originates from the 'CIS 90 Test 1 (practice)' entry and points to the detailed quiz page shown in the next screenshot.

The screenshot shows the detailed view of the 'CIS 90 Test 1 (practice)' quiz. The page header includes the course name 'CIS-90 (Room 828)' and the quiz title. The quiz details are as follows:

Due	Points	Questions
Mar 2 at 10:30am	33	33

Additional details include: Available Feb 24 at 7am - Mar 2 at 10:30am (7 days), Time Limit 60 Minutes, and Allowed Attempts Unlimited.

INSTRUCTIONS:

Every question on the test was designed to be answered using one of the systems below.

- oslab cis.cabrillo.edu (port 2220) - This server is named Opus internally.
- sun-hwa-iv cis.cabrillo.edu (port 22)
- daughter-of-opus simms-teach.com (port 2222)
- arya-xx (port 22) - Select xx for your own Arya.

Each question begins with [system name] so you know which system you should be logged into to answer the question.

All systems are accessible using ssh from opus. For sun-hwa-iv and daughter-of-opus login using your original opus credentials. For arwen, use the generic cis90 account.

IF YOU GET STUCK ON A QUESTION you can ask your classmates for help on the forum. On the real test you can ask the instructor for the answer and forfeit the points. For the real test the instructor will be available during the class and online between 8-10 PM in the evening for online and long distance students.

Please KEEP YOUR ANSWERS TO A SINGLE LINE ONLY!!

A full-page background image showing a sunset over a beach. The sky is filled with vibrant orange, pink, and purple clouds. The sun is low on the horizon, casting a warm glow. To the right, a dark, silhouetted cliff rises from the beach. The foreground shows the wet sand of the beach reflecting the colors of the sky, with some dark rocks scattered about.

Wrap up

New commands:

NA

NA

New metacharacters:

?

Matches any single character

[]

Matches any character in the brackets

New Files and Directories:

NA

NA

Next Class

Assignment: Check Calendar Page on web site to see what is coming up.

No Quiz
No Lab due
Test !

End Meeting

End Meeting

Backup