



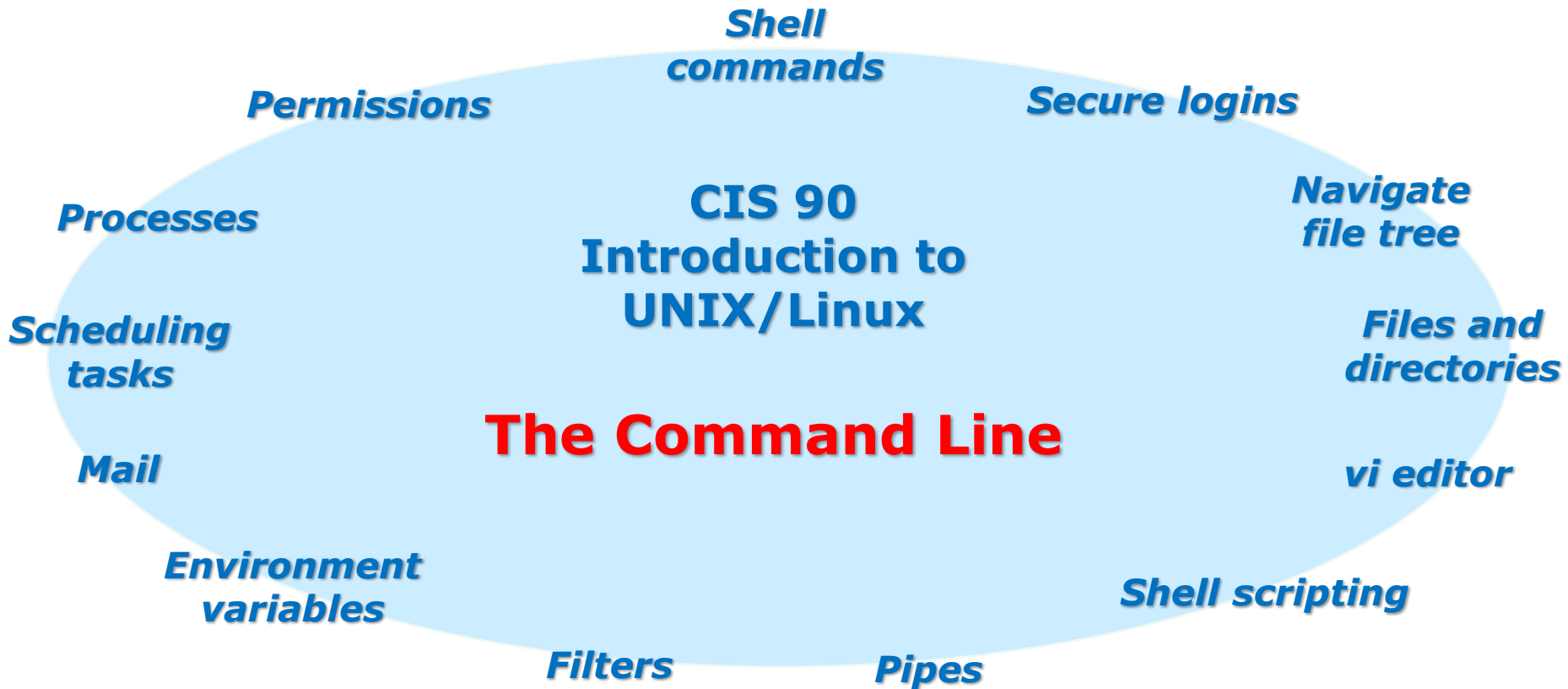
Rich's lesson module checklist

Last updated 10/24/2018

- 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
- Schedule lock of turnin directory and submit
scripts/schedule-submit-locks
- Opus-II - hide script tested
- Update test Q21 for number of accounts (Sp18 /var "lost" two city files - **FIX**)
- Practice test available on Canvas at end of class
- P2 test system online and unlocked at end of class
- Flash cards, teams and timer script ready
- Clean up mysql database
- 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: <https://web.archive.org/web/20140209023942/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. John's site: <http://teacherjohn.com/>
- Jaclyn Kostner for many webinar best practices: e.g. mug shot page.



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, the "CIS 90" link is highlighted. The main content area shows the "CIS 90 (Fall 2014) Calendar" with a "Calendar" link highlighted. Below this is a table with columns for "Lesson", "Date", "Topics", and "Links". The table contains the following information:

Lesson	Date	Topics	Links
		Class and Lab Overview <ul style="list-style-type: none"> Understand how the course will work High-level overview of computers, operating systems, and virtual machines Overview of LINUX/Linux market and architecture Using SSH for remote network exits Using terminals and the command line 	
	9/2	Methods Presentation slides (download)	(day)
		Supplemental <ul style="list-style-type: none"> PowerPoint: Logging into Opus (download) 	2.4.5 p163-172 p164-172 (pdf)
		Assignments <ul style="list-style-type: none"> Student Survey Lab 1 	
		CCS Center Enter virtual classroom	
		Quiz 1	
		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.

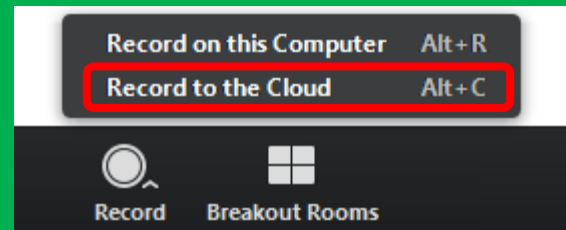
The screenshot shows a Zoom meeting interface with several windows open. The main window displays a login page for 'Rich's Cabrillo College CIS 90' with the text 'Get into the car' overlaid. Other windows include the Google search page, the CIS 90 website's 'CIS 90 Calendar' page, and a PDF of lesson slides titled 'CIS 90 - Lesson 1' with the subtitle '90 System Playground'. The Zoom meeting controls at the bottom show '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

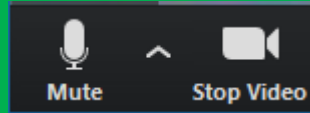


Start



Start Recording

Audio Check



Start Recording

Audio & video Check



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



Mikey



Jona



Joseph



Tara Marie



Fredi



Carina



Isaac



Matthew



Erik



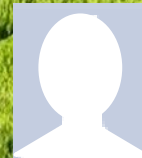
Tony



Branden



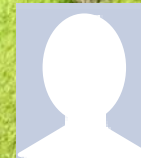
Dominic



Ryan L.



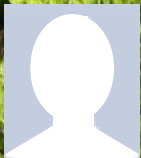
Alejandra



Blair



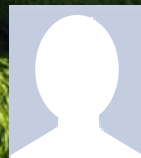
Zari



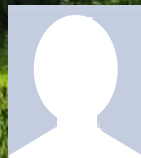
Victor



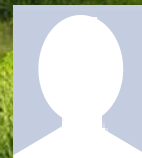
Danny



Gabriel



Janelly



Austin

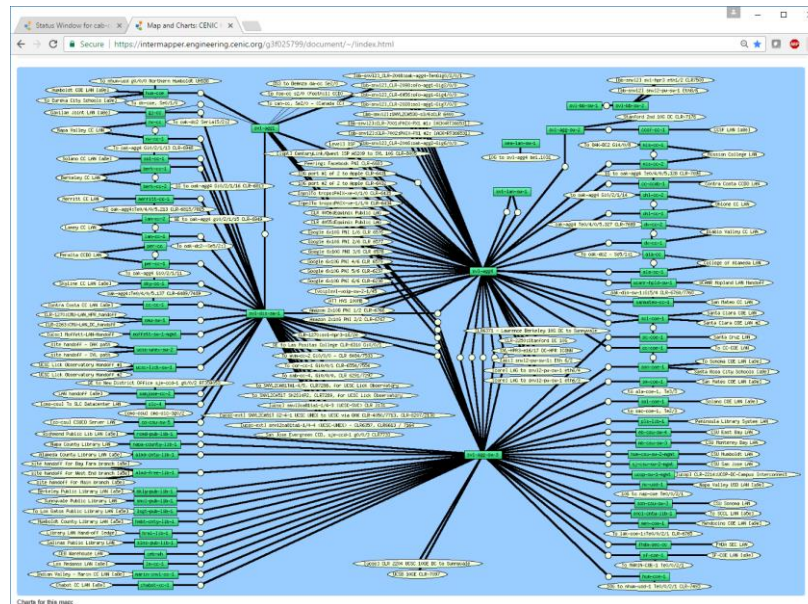


Aaron



Ryan M.

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 ConferZoom White Board

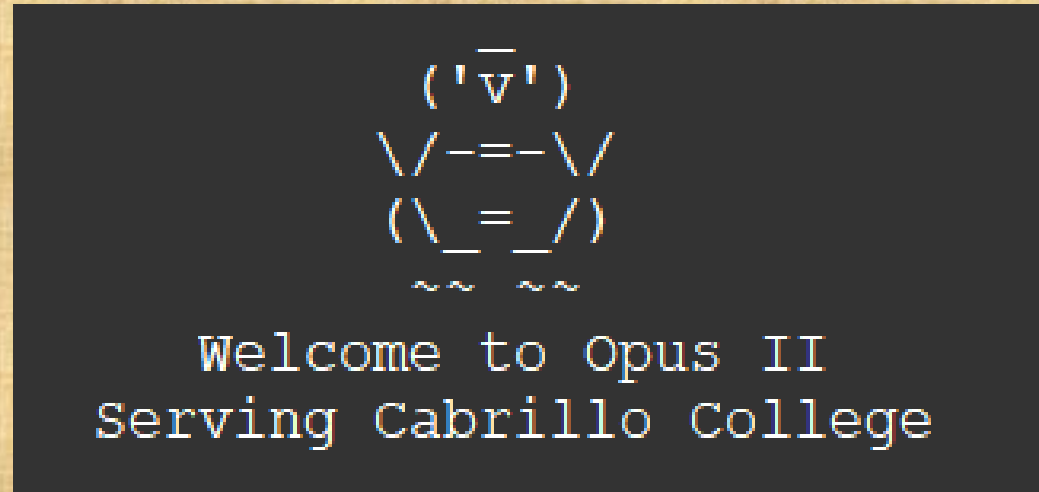
email answers to: risimms@cabrillo.edu

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

Review

Objectives	Agenda
<ul style="list-style-type: none">• Get ready for the next test• Practice skills• Introduction to processes	<ul style="list-style-type: none">• Quiz• Questions• Housekeeping• Linux at school• Linux at home• More on I/O• All together now• Subtle differences• Errors• 2>&1• More on I/O - programming• umask• More pipeline practice• Pipeline and redirection practice• More on pipelines• Eggs, treats and tricks• Review• Make teams• Flashcard practice• Assignment• Wrap up

Class Activity



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

Class Activity

Unit 3

Electronic Mail

- Guest operator: Enable Work on DTC (On-The-Job) training programs
- Learn how to use the LINC 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

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

If you haven't already,
download the lesson slides

Class Activity

	<ul style="list-style-type: none">• Read/skim Lesson 1 slides• Student Survey• Lab 1
	ConferZoom <ul style="list-style-type: none">• Enter virtual classroom• Class archives
	Quiz 1
	Commenda <ul style="list-style-type: none">• 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? Tests?

How this course works?

- Graded work & tests in home directories
- Answers in /home/cis90/answers

Who questions much, shall learn much, and retain much.

- Francis Bacon

If you don't ask, you don't get.

- Mahatma Gandhi

Chinese
Proverb

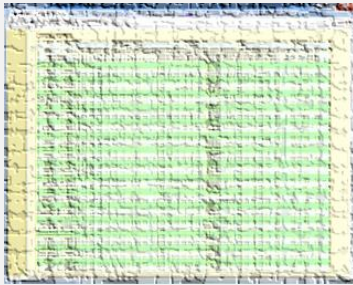
他問一個問題，五分鐘是個傻子，他不問一個問題仍然是一個傻瓜永遠。

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

Review your progress in the course

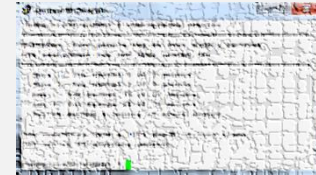
Check the website Grades page

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



Or check on Opus-II

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



Written by Jesse Warren a past CIS 90 Alumnus

- **Send me your survey to get your LOR codename.**
- **Graded labs and tests are in your home directories.**

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

Points that could have been earned:

6 quizzes: 18 points
 6 labs: 180 points
 1 test: 30 points
 2 forum quarters: 40 points
Total: 268 points

Extra Credit

On the forum

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

In lesson slides
(search for extra credit)

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`



On the website

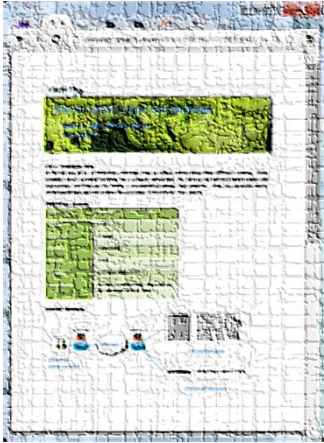
<http://simms-teach.com/cis90grades.php>

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)

Lab Assignments -- Pearls of Wisdom



- Don't wait till the last minute to start.
- Plan for things to go wrong and give yourself time to ask questions and get answers.
- The *slower* you go the *sooner* you will be finished.
- A few minutes reading the forum can save you hour(s).
- Line up materials, references, equipment and software ahead of time.
- It's best if you fully understand each step as you do it. Use Google or refer back to lesson slides to understand the commands you are using.
- Keep a growing cheat sheet of commands and examples.
- Study groups are very productive and beneficial.
- Use the forum to collaborate, ask questions, get clarifications and share tips you learned while doing a lab.
- **Late work is not accepted** so submit what you have for partial credit.

Getting Help When Stuck on an Assignment

- Google the topic/error message.
- Search the Lesson Slides (they are PDFs) for a relevant example on how to do something.
- Check the forum. Someone else may have run into the same issue and found a way past it. If not start a new topic, 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:

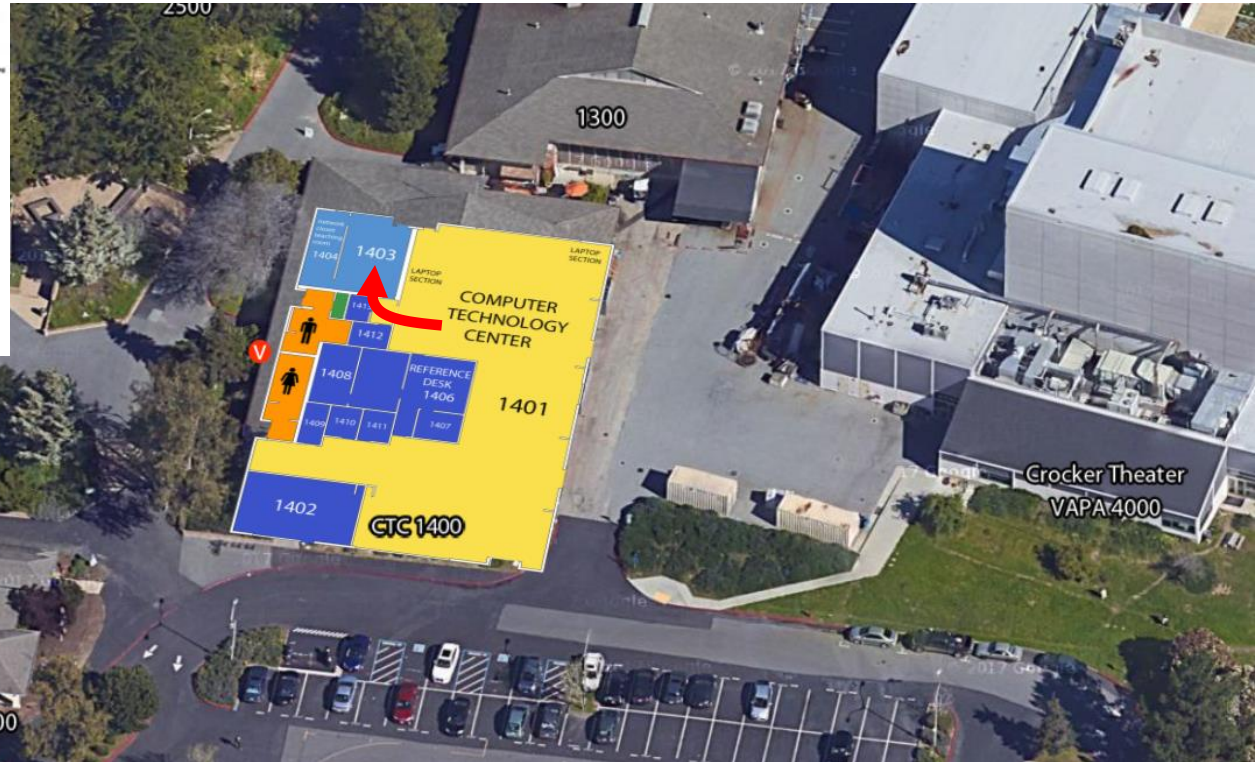
<https://www.cabrillo.edu/salsa/listing.php?staffId=1426>

I'm in the CTC (room 1403) every Tuesday from 3:30-5:00 pm.

- 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!

CIS Labs always involve some troubleshooting!

CTC - Building 1400 On lower campus



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

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 & 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	Mon 1/29	Tue 1/30	Wed 1/31	Thu 2/1
10am				
11am				
12pm				
1pm				
2pm	1:15p - 3p Jeffrey Bergamini CS Instructor Carter Frost CIS/CS	1:40p - 5p Jeffrey Bergamini at CS Carter Frost CIS/CS	1:15p - 3p Jeffrey Bergamini CS Instructor Carter Frost CIS/CS	1:40p - 5p Jeffrey Bergamini at CS 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



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 ...
- 5) If you had trouble doing the last test ...

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

Email: risimms@cabrillo.edu

Housekeeping





Pause/Stop Recording

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



Resume/Stop Recording

Resume Recording

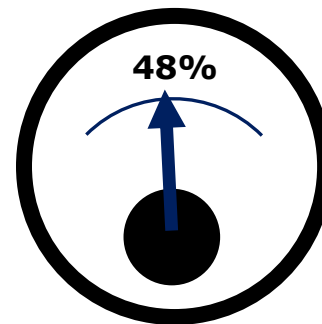
Audio Check

CIS Fundraising "Bake Sale"

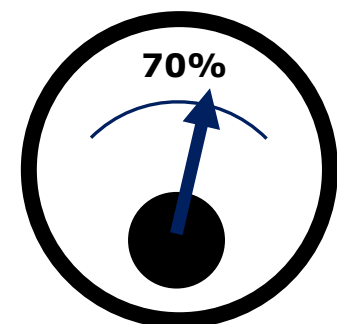
LAST CHANCE

*Donate by answering
seven yes/no questions
on an online
Perkins/VTEA survey!*

Last Week



This Week



Perkins/VTEA Survey

LAST CHANCE

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 September 27, 2015. The text of the post explains that the Carl D. Perkins Vocational and Technical Education Act was originally authorized by Congress in 1966 and reauthorized in 1991 and again in 2009. It provides federal funding for improving career technical education (CTE) within the United States to help the economy. The post states that Cabrillo College is receiving a portion of this funding and is asking students to complete a survey. The survey is available online at <http://www.cabrillo.edu>. The post also includes a search bar, a user profile for Rich Simms, and a list of search terms.

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!**

Even if you took the survey in another CIS class!

Career Technical Information
Your answers to these questions will help qualify Cabrillo College for Perkins/VTEA grant funds.

Are you currently receiving benefits from:

TANF/CALWORKS
 Yes
 No

SSI (Supplemental Security Income)
 Yes
 No

GA (General Assistance)
 Yes
 No

Does your income qualify you for a fee waiver?
 Yes
 No

Are you a single parent with custody of one or more minor children?
 Yes
 No

Are you a displaced homemaker attending Cabrillo to develop job skills?
 Yes
 No

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?
 Yes
 No

Housekeeping

1. Lab 7 due 11:59PM tonight -- **don't forget to submit your latest version!**
2. Read your Opus-II email for Lab 7 submission status.
3. A **check7** script is available.
4. Fine Print:
Test #2 is scheduled for our next class!




Test #2 will happen during our next class!

Practice test available after class.



**Test #2 is scheduled
for our next next
class!**

**Practice test
available after class.**



**Test #2 is scheduled for
our next class!**

**Practice test available
after class.**

Test #2

1. Test #2 is **scheduled for our next class!** Same format as before. The test will start during the last hour of class. If you work you can take it later in the day as long as it is completed by 11:59PM.
2. Practice Test #2 is available after class on Canvas!
3. Work the Practice Test BEFORE the real test begins.
4. The Practice Test and Practice Test server will be available until about 30 minutes before the real test starts.

How to pass Test #2 with flying colors

- Keep taking the practice test until you can complete each question in 30 seconds or less. Use the **./restore** command to restore your directory on the test system to the original state.
- Use the forum to discuss your approaches and results with classmates.
- If a question takes longer than 30 seconds ask for help. You can ask for help on the forum, see me during office hours, work with a tutor in the STEM center, join a study group or all of the above!
- Create a custom crib sheet of commands and key concepts covered in the course.
- Use the flashcards on the course website and rework any labs and previous tests you want to better understand.
- Tip: Use the **-v** option on **chmod**, **mv**, **cp** and **rm** commands to see what actually happened.
- But most important ... DON'T WAIT TILL THE LAST MINUTE to prepare!

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 Fall 2014 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 Fall 2014 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.*

FALL 2018 FINAL EXAMINATIONS SCHEDULE DECEMBER 10 TO DECEMBER 15

DAYTIME FINAL SCHEDULE

Daytime Classes: All times in bold refer to the beginning times of classes. **MW/Daily** means Monday alone, Wednesday alone, Monday and Wednesday or any 3 or more days in any combination. **TTH** means Tuesday alone, Thursday alone, or Tuesday and Thursday. **Classes meeting other combinations of days and/or hours not listed must have a final schedule approved by the Division Dean.**

STARTING CLASS TIME / DAY(S)	EXAM HOUR	EXAM DATE
<i>Classes starting between:</i>		
6:30 am and 8:55 am, MW/Daily	7:00 am-9:50 am	Monday, December 10
9:00 am and 10:15 am, MW/Daily	7:00 am-9:50 am	Wednesday, December 12
10:20 am and 11:35 am, MW/Daily	10:00 am-12:50 pm	Monday, December 10
11:40 am and 12:55 pm, MW/Daily	10:00 am-12:50 pm	Wednesday, December 12
1:00 pm and 2:15 pm, MW/Daily	1:00 pm-3:50 pm	Monday, December 10
2:20 pm and 3:35 pm, MW/Daily	1:00 pm-3:50 pm	Wednesday, December 12
3:40 pm and 5:30 pm, MW/Daily	4:00 pm-6:50 pm	Wednesday, December 12

CIS 90

Introduction to UNIX/Linux

Provides a technical overview of the UNIX/Linux operating system, including hands-on experience with commands, files, and tools. Recommended Preparation: CIS 1L or CIS 72.

Transfer Credit: Transfers to CSU;UC

Section	Days	Times	Units	Instructor	Room
1	W	1:00PM-4:05PM	3.00	R.Simms	OL
&	Arr.	Arr.		R.Simms	OL
Section 1 is an ONLINE course. Meets weekly throughout the semester online during the scheduled times by remote technology with an additional 50 min online lab per week. For details, see instructor's web page at go.cabrillo.edu/online . This course has zero cost for textbooks.					
2	W	1:00PM-4:05PM	3.00	R.Simms	828
&	Arr.	Arr.		R.Simms	OL
Section 2 is a Hybrid ONLINE course. Meets weekly throughout the semester at the scheduled times with an additional 50 min online lab per week. For details, see instructor's web page at go.cabrillo.edu/online . This course has zero cost for textbooks.					

Heads up on Final Exam

Test #3 (final exam) is **MONDAY December 10th 1-3:50PM**

Mon	12/10	Test #3 (the final exam)	<u>5 posts</u> <u>Lab X1</u> <u>Lab X2</u>
		<p>Time</p> <ul style="list-style-type: none"> MONDAY 1:00PM - 3:50PM in Room 828 or online <p>Materials</p> <ul style="list-style-type: none"> Presentation slides (<u>download</u>) Test (<u>canvas</u>) <p>ConferZoom</p> <ul style="list-style-type: none"> <u>Enter virtual classroom</u> <u>Class archives</u> 	

*Extra credit
labs and final
posts due by
11:59PM*

- All students will take the test at the same time. The test must be completed by **3:50PM**.
- Working and long distance students can take the test online via ConferZoom and Canvas.
- Working students will need to plan ahead to arrange time off from work for the test.
- Test #3 is **mandatory** (even if you have all the points you want)

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 a web browser window displaying the NDG Linux Essentials website. The page title is "2.3 Major Open Source Applications". The content includes an introduction to Linux software, a list of categories (Server software, Desktop software, Tools), and a terminal window showing the Ubuntu login process. The terminal output includes: "Starting domain name service... bind9", "Welcome to Ubuntu 12.04.5 LTS (GNU/Linux 3.17.7 x86_64)", "Documentation: https://help.ubuntu.com/", "This lab has two user accounts (username :: password)", "root :: netlab22", "student :: netlab22", "Press the [Enter] key to begin...", and "The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/*-copyright. Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law."

<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.



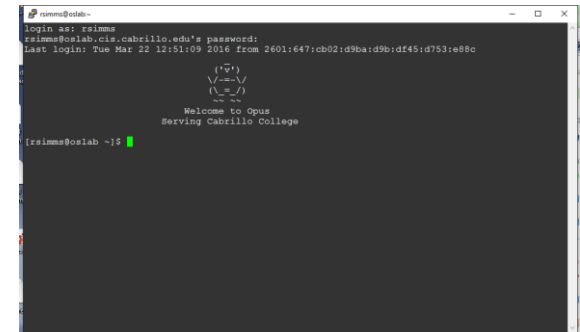
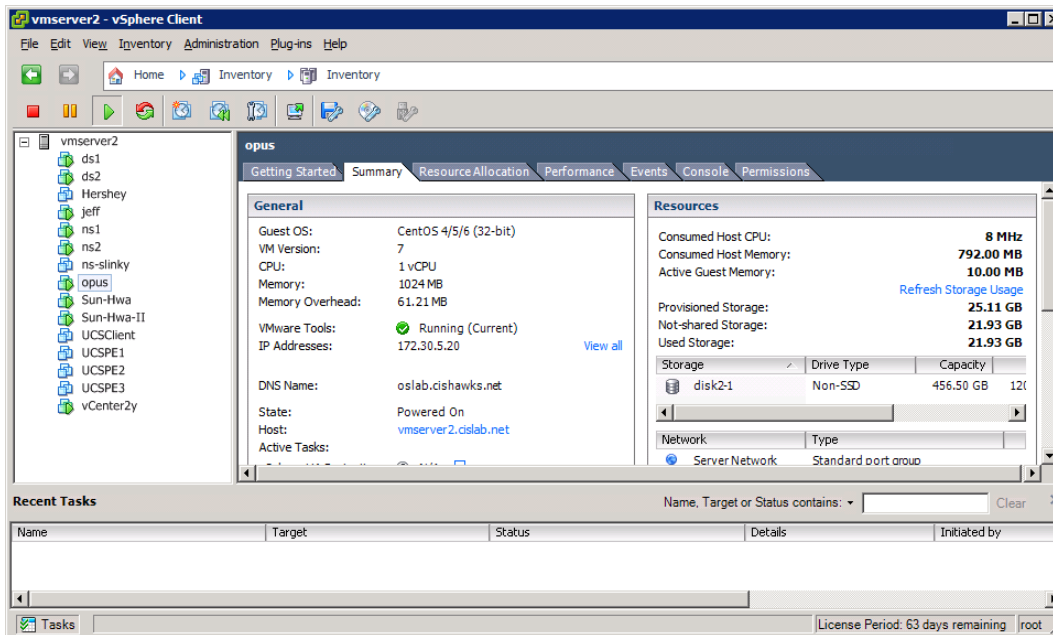
Linux at School

Our Opus-II server on campus

Dell R610 Server



VMware vSphere Client



Opus-II is a VM running on one of the VMware ESXi servers in the CIS Datacenter

Linux at Home

USB "Live" Linux Boot USB Drive

Allows you to use or try out Linux on an existing computer without installing it

1)



Get the Linux distros of your choice
Or see: <http://iso.linuxquestions.org/>

2)



Get a USB flash drive

Google "boot live linux from usb" for instructions

3)

Or see: <http://www.pendrivelinux.com/yumi-multiboot-usb-creator/>



4)



Configure your BIOS to boot from USB then select the Operating System as your computer boots up

USB "Live" Linux Multi-Boot USB Flash Drive

Windows



CentOS



Kali



YUMI formatted Flash Drive
(www.pendrivelinux.com)

Linux Mint

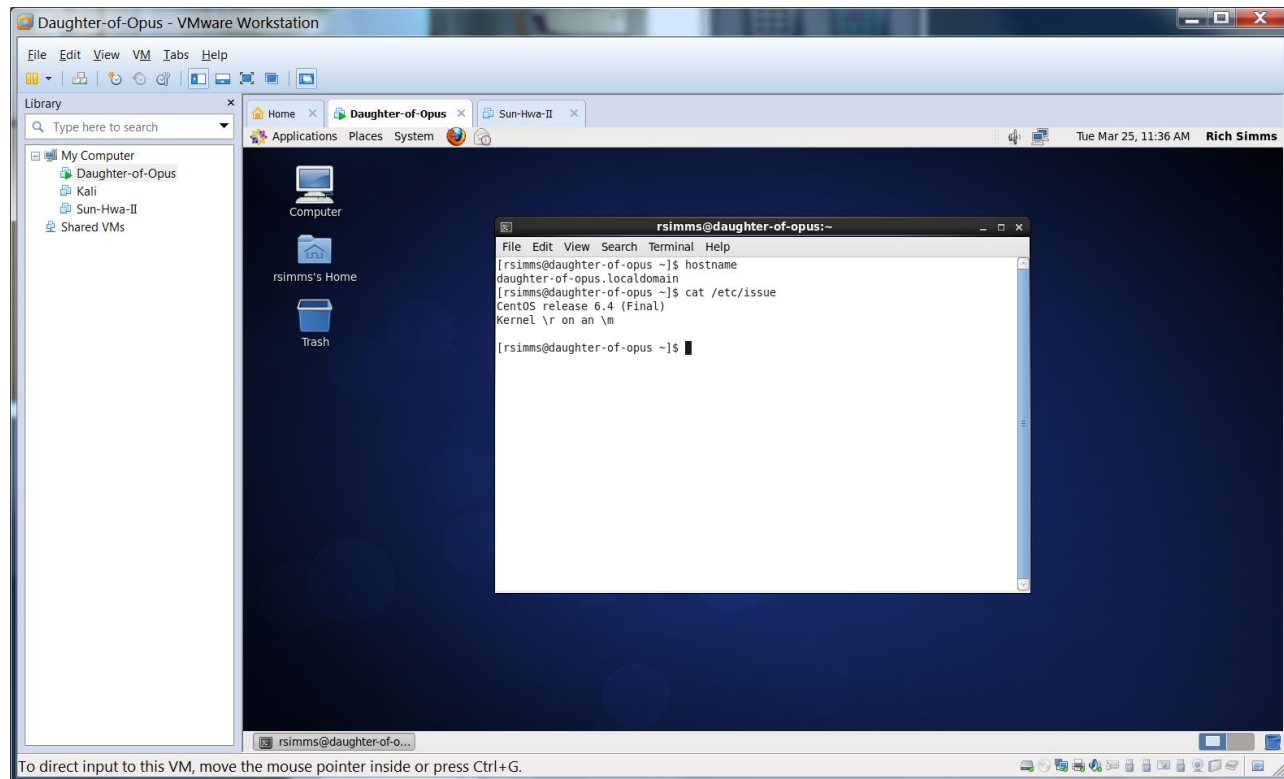


Ubuntu



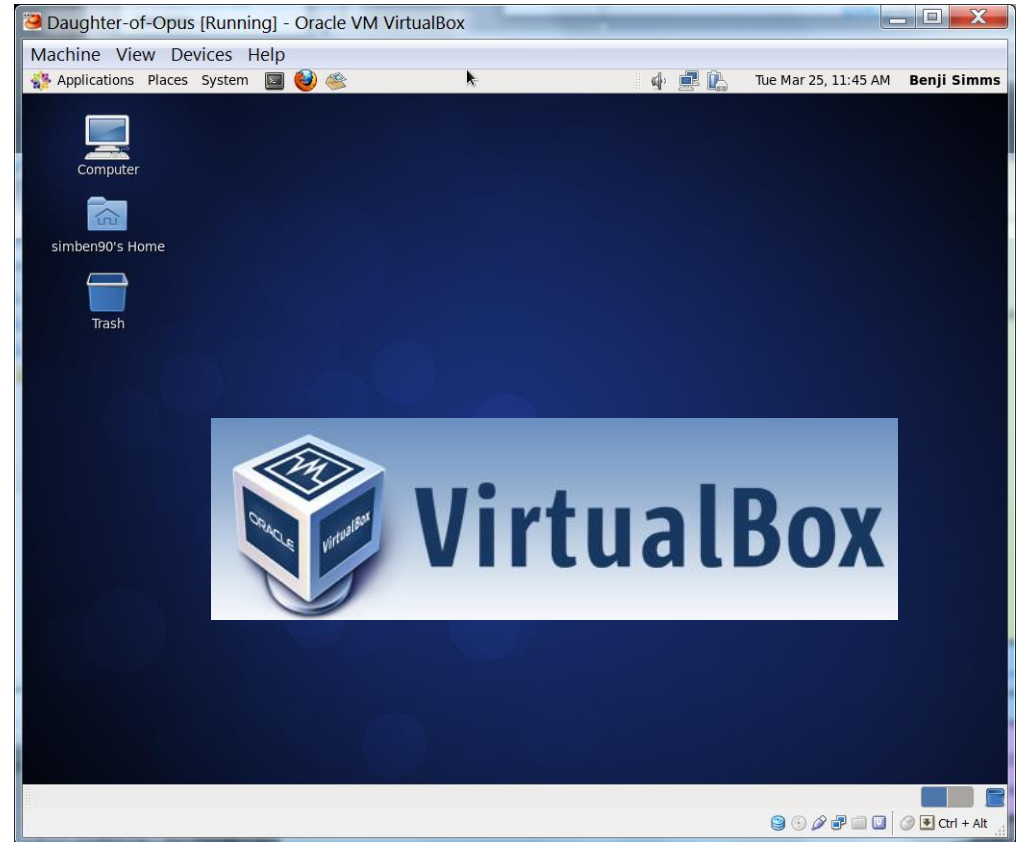
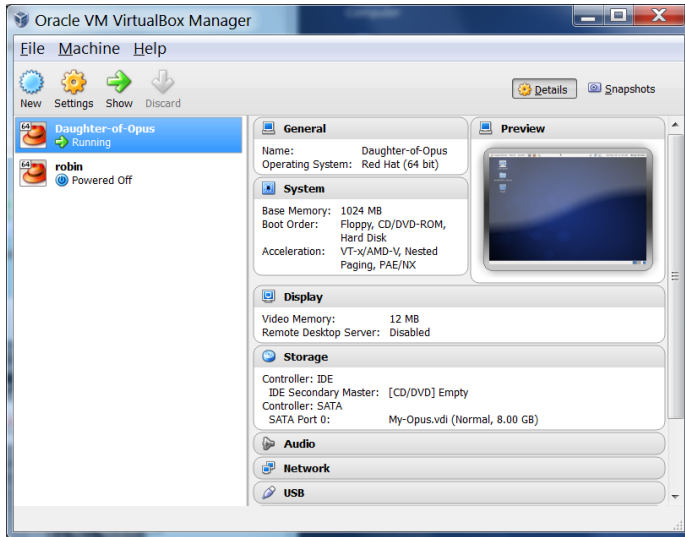
Allows you to use or try out Linux on an existing computer without installing it

VMware Workstation (PC) or Fusion (Mac)



One Daughter-of-Opus is a VM running on my laptop using VMware Workstation (expires in one year)

VirtualBox



This Daughter-of-Opus is a VM running on my laptop using Oracle VirtualBox (never expires)

Amazon Web Services



The screenshot shows the Amazon EC2 console interface. The main content area displays a table with one instance: 'Son-of-Opus' (Instance ID: i-6bf57f31, Instance Type: t1.micro, Availability Zone: us-west-1a, Instance State: running). Below the table, the details for the selected instance are shown, including its Public DNS, Elastic IP, Private DNS, Private IPs, Availability zone, Security groups, Scheduled events, AMI ID, and Subnet ID.

Instance: i-6bf57f31 (Son-of-Opus)	
Instance ID	i-6bf57f31
Instance state	running
Instance type	t1.micro
Availability zone	us-west-1a
Security groups	quick-start-1, view rules
Scheduled events	No scheduled events
AMI ID	RHEL-6_4_CA-x86_64-10-Hourly2 (ami-6283a827)
Public DNS	ec2-54-215-232-67.us-west-1.compute.amazonaws.com
Elastic IP	-
Private DNS	ip-172-31-3-240.us-west-1.compute.internal
Private IPs	172.31.3.240
Secondary private IPs	-
VPC ID	vpc-4fdeed27
Subnet ID	subnet-41dedd29

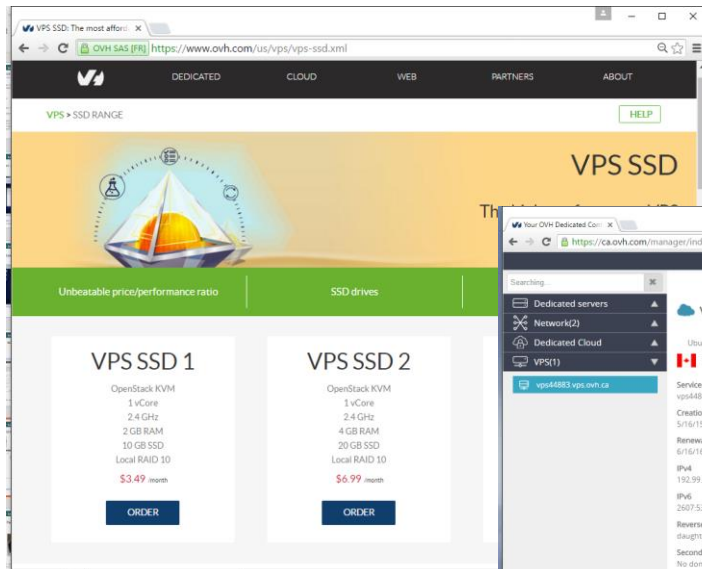


Son-of-Opus is a VM running on Amazon Web Services

Single VM is free for a year, then about \$60 per month after that.

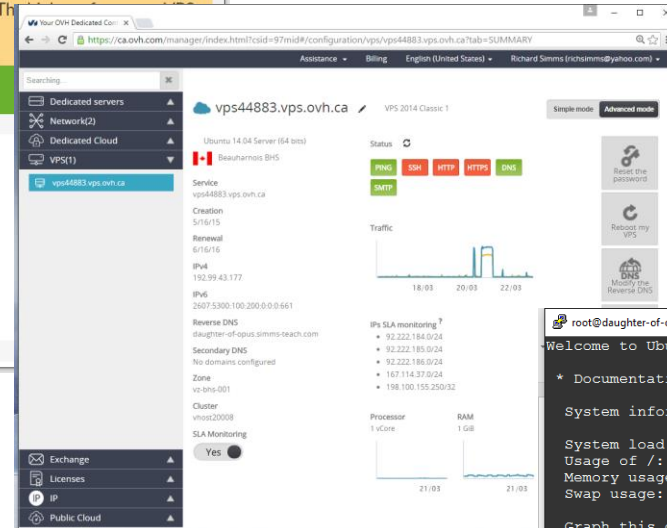
OVH.com

Purchase



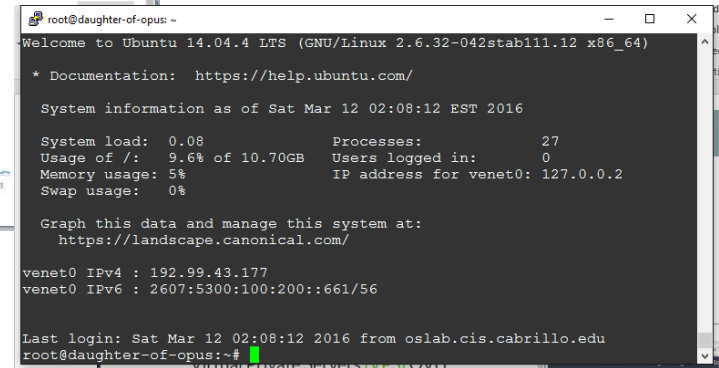
The screenshot shows the OVH.com website's 'VPS SSD' page. It features a navigation bar with 'DEDICATED', 'CLOUD', 'WEB', 'PARTNERS', and 'ABOUT'. Below the navigation, there's a header 'VPS > SSD RANGE' and a 'HELP' button. The main content area is titled 'VPS SSD' and includes a sub-header 'Unbeatable price/performance ratio' and 'SSD drives'. Two product cards are displayed: 'VPS SSD 1' and 'VPS SSD 2'. Both are based on OpenStack KVM and offer 1 vCore, 2.4 GHz, and 4 GB RAM. VPS SSD 1 has 2 GB RAM and costs \$3.49/month. VPS SSD 2 has 20 GB SSD and Local RAID 10, costing \$6.99/month. Both cards have an 'ORDER' button.

Configure



The screenshot shows the OVH.com management console for a VPS. The URL is <https://ca.ovh.com/manager/index.html?csid=97mid#/configuration/vps/vps44883.vps.ovh.ca?tab=SUMMARY>. The page displays details for 'vps44883.vps.ovh.ca' (VPS 2014 Classic 1). It shows the service is 'Ubuntu 14.04 Server (64 bits)' on 'Beauharnois BHS'. The status is 'ON', with 'SSH', 'HTTP', 'HTTPS', and 'DNS' all active. Traffic graphs show activity between 18:03 and 22:03. The console also displays IP addresses (IPv4: 192.99.43.177, IPv6: 2607:5300:100:200:0:0:661/56), reverse DNS, and SLA monitoring options.

Use



```
root@daughter-of-opus: ~  
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 2.6.32-042stab111.12 x86_64)  
  
* Documentation: https://help.ubuntu.com/  
  
System information as of Sat Mar 12 02:08:12 EST 2016  
  
System load: 0.08          Processes:           27  
Usage of /:  9.6% of 10.7GB Users logged in:    0  
Memory usage: 5%          IP address for venet0: 127.0.0.2  
Swap usage:  0%  
  
Graph this data and manage this system at:  
https://landscape.canonical.com/  
  
venet0 IPv4 : 192.99.43.177  
venet0 IPv6 : 2607:5300:100:200::661/56  
  
Last login: Sat Mar 12 02:08:12 2016 from oslab.cis.cabrillo.edu  
root@daughter-of-opus:~#
```

Virtual private servers like daughter-of-opus used on Test #1 costs \$3 per month

Small Form Factor Servers



HP Microserver



https://www.hpe.com/us/en/product-catalog/servers/proliant-servers.filters-facet_subbrand_url:ProLiant-MicroServer.hits-12.html



SuperMicro



<https://tinkertry.com/my-tinkertry-d-xeon-d-bundle-2-supermicro-superserver-bundle-2-of-joy>

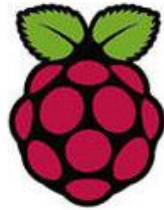
Comparatively inexpensive "bare bones" servers that come without memory, hard drives or an operating system

Fantastic Bargains on EBay

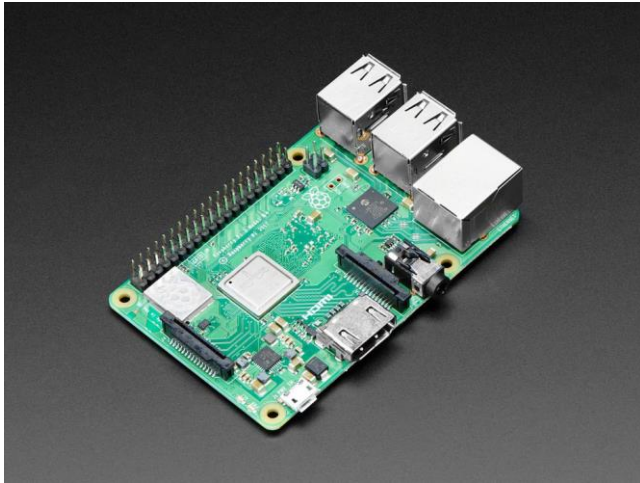
The screenshot shows an eBay search results page for 'Dell R610 Servers'. The browser address bar shows 'www.ebay.com/bhp/dell-r610'. The page header includes the eBay logo, a search bar, and navigation links. The main content area displays three product listings:

- Listing 1:** Dell PowerEdge R610 Virtualization Server 2.53GHz 8-Core E5540 32GB 2x146G PERC6. Price: \$274.30. Condition: Seller refurbished. Time left: 28d 15h 47m. Location: Georgia. Sold by: savemyserver (21342). Includes a 'Top Rated Plus' badge.
- Listing 2:** Dell Poweredge R610 2x Xeon E5506 2.13ghz Quad Core / SAS 6ir 2PS Add RAM. Price: \$109.99. 24 watching | 17 sold.
- Listing 3:** Dell PowerEdge R610 Virtualization Server X5570 2.93GHz 8-CORES 48GB 2x 146GB. Price: \$274.99. 45 watching | 19 sold.

The left sidebar contains navigation options such as 'View all Dell', 'Dell R610 Memory', 'Browse Related' (listing other Dell server models), and 'Also shop in' (listing categories like Computers/Tablets & Networking).

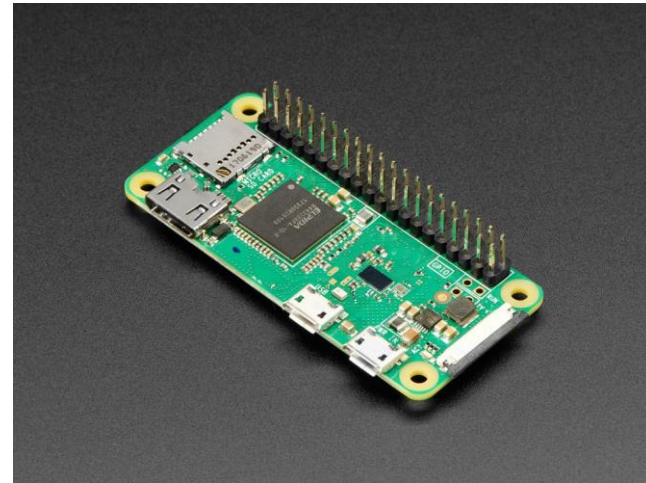


Raspberry Pi



Raspberry Pi 3 Model B+

<https://www.adafruit.com/product/3775>



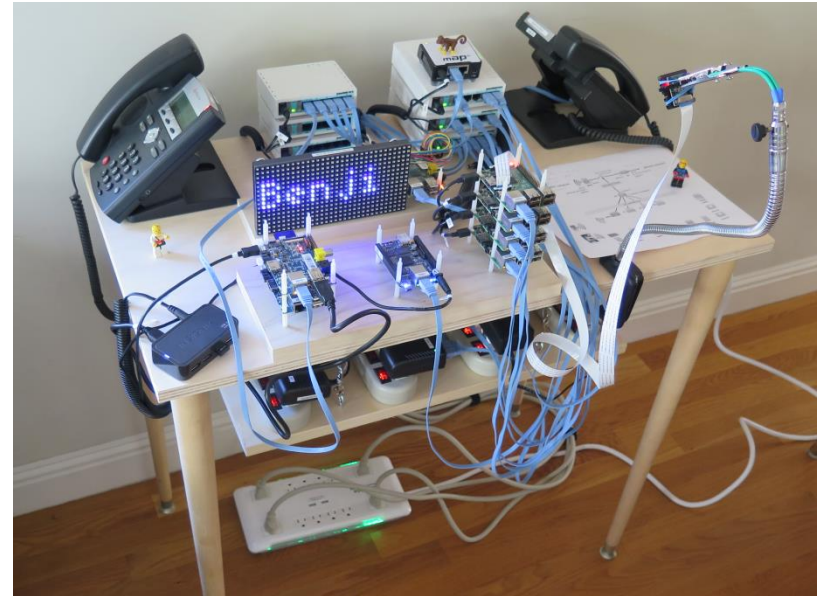
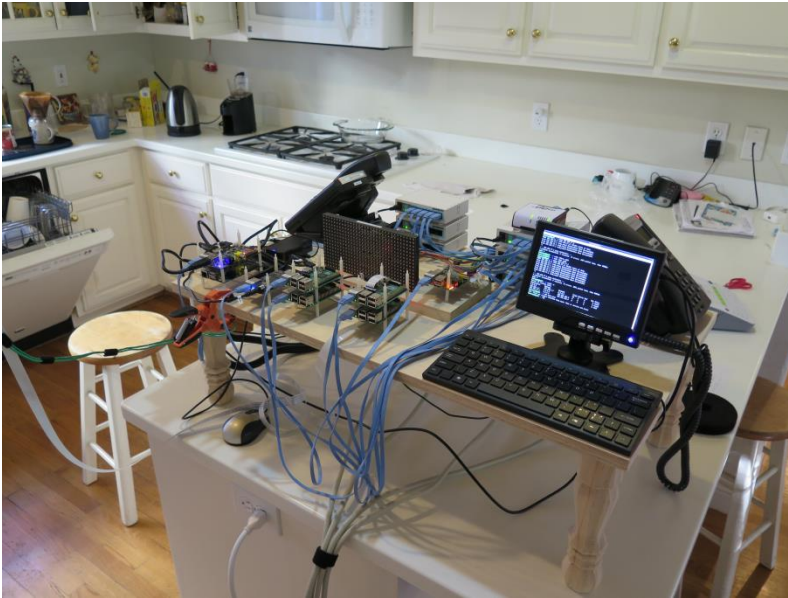
Raspberry Pi Zero WH

<https://www.adafruit.com/product/3708>



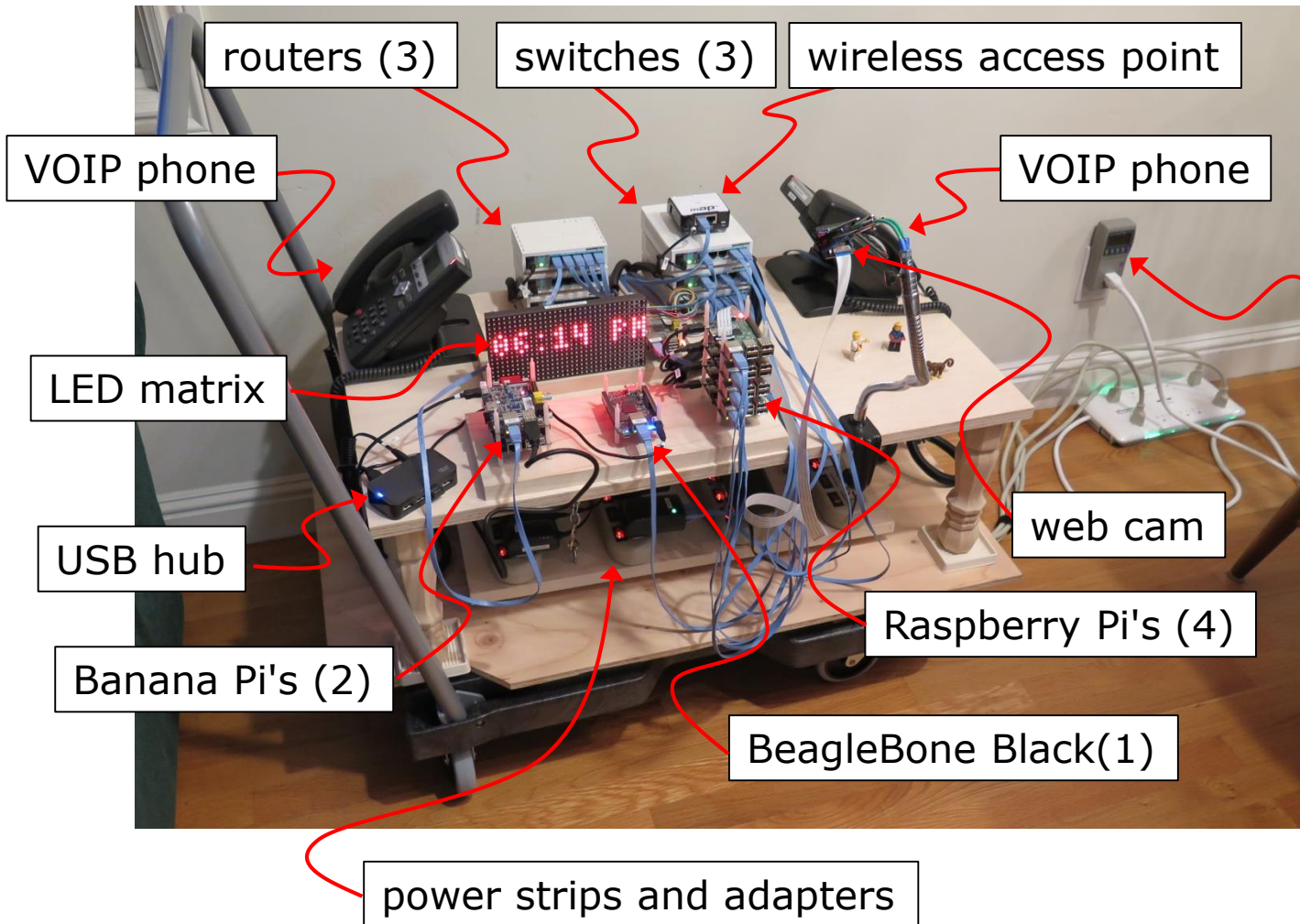
Micro Datacenters

A very tiny home made datacenter



Low cost way to practice all your CIS network and system skills

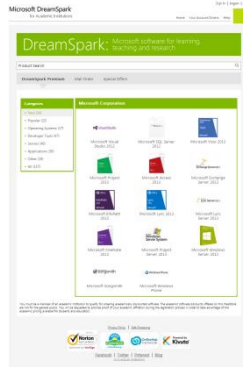
A very tiny home made datacenter



39 watts total

See "Your Name in Lights" post in forum

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



Microsoft
Software
(Academic)

VMware
Software
(Academic)



Rich's Cabrillo College CIS Classes Resources

Home
Resources
Forums
CIS Lab
Blackboard

Login

Flashcards

Admin

CIS 90

Previous Classes

60 days - term ends!

Cabrillo College

Web Advisor

Commands and Files

VLab RDP file

CIS 90 VLab VM Assignments

RIP Dennis Ritchie

Opus Status: UP

Links

<p>Instructors</p> <ul style="list-style-type: none"> Linux Master Jim Programming Master Ed Network Master Gerinde Network Master Rick Web Master John Systems Master Michael <p>Clubs</p> <ul style="list-style-type: none"> Computer Club Robotics Club <p>Departments</p> <ul style="list-style-type: none"> CNSA CIS CS <p>Crib Sheets</p> <ul style="list-style-type: none"> Olle Wright (CIS 90) <p>Documentation</p> <ul style="list-style-type: none"> TLDP LINFO UNIX Rosetta Stone <p>Animations</p> <ul style="list-style-type: none"> Linux network technologies 	<p>Getting Linux/UNIX</p> <ul style="list-style-type: none"> Linux ISOs Kernels RPMS (rpmfind) RPMS (pbone) OpenSolaris <p>Tools and Software</p> <ul style="list-style-type: none"> Apache Bastille CoRD cygwin DOS boot disks Dynamics/Dynagen John the Ripper Netfilter Putty SSH Tools Quagga routing suite Tripwire Wireshark <p>Academic Software for CIS Students</p> <ul style="list-style-type: none"> Microsoft Webstore VMware Webstore <p>Virtualization</p> <ul style="list-style-type: none"> VirtualBox VMware ESXi and vSphere client <p>Standards</p> <ul style="list-style-type: none"> IEEE IETF (RFCs) <p>Training and Tutorials</p> <ul style="list-style-type: none"> Free Linux Tutorials The Linux Foundation Linux Survival Learn about Linux The Linux Tutorial 	<p>Commands</p> <ul style="list-style-type: none"> Practical Command Directory Useful vi summary vi cheat sheet <p>Howtos</p> <ul style="list-style-type: none"> HowtoForge email DNS Ethernet (NIC drivers) NES NIS PPP Putty SSH Keys Using sed <p>Student Howtos</p> <ul style="list-style-type: none"> Monitor Script by Sean Calahan WiFi Penetration by Ryan Scher Logging into Opus from a Mac by Laura Sreckovic LDAP Implementation by Tim Childers Install and DualBoot into Microsoft Windows 7 and Linux Ubuntu by Richie Fou Making an ethernet cable by Michael George Home VM access via Linksys router by Marc Romansky Putty to VMs by Marc Romansky Installing VirtualBox by Marcos Valdebenito Linux Permissions by Michael Wicherski Guide to /bin/mail by Michael Wicherski <p>Linux News</p>
---	---	---

Linux Distros (ISOs)

VirtualBox

70

More on I/O

(input/output)

Input and Output

File Redirection

The 3 standard UNIX file descriptors:

Name	Integer Value
stdin (st andard in)	0
stdout (st andard out)	1
stderr (st andard error)	2

*Every process is provided with three file descriptors: **stdin**, **stdout** and **stderr***

Input and Output

File Redirection

The input and output of a program can be **redirected** to and from other files as follows:

0< *filename*

Redirects **stdin**, input will now come from *filename* rather than the keyboard.

1> *filename*

Redirects **stdout**, output will now go to *filename* instead of the terminal.

2> *filename*

Redirects **stderr**, error messages will now go to *filename* instead of the terminal.

>> *filename*

Redirects **stdout**, output will now be appended to *filename*.

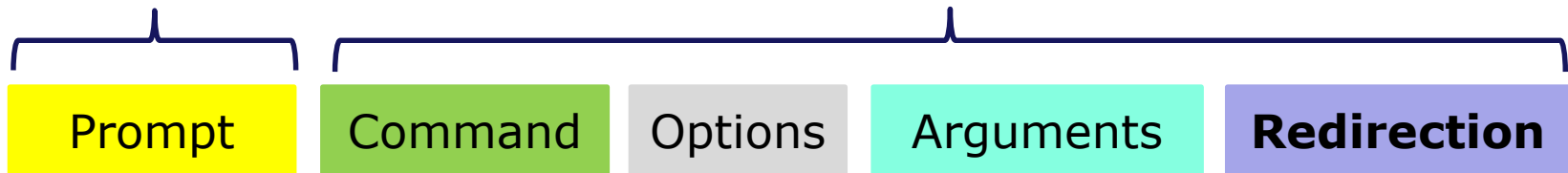
2>> *filename*

Redirects **stderr**, output will now be appended to *filename*.

The redirection is specified on the command line

Shell prints this
to prompt user to
enter a command

Shell parses this command line



Redirection connects **stdin**, **stdout** and **stderr** to non-default devices

Examples

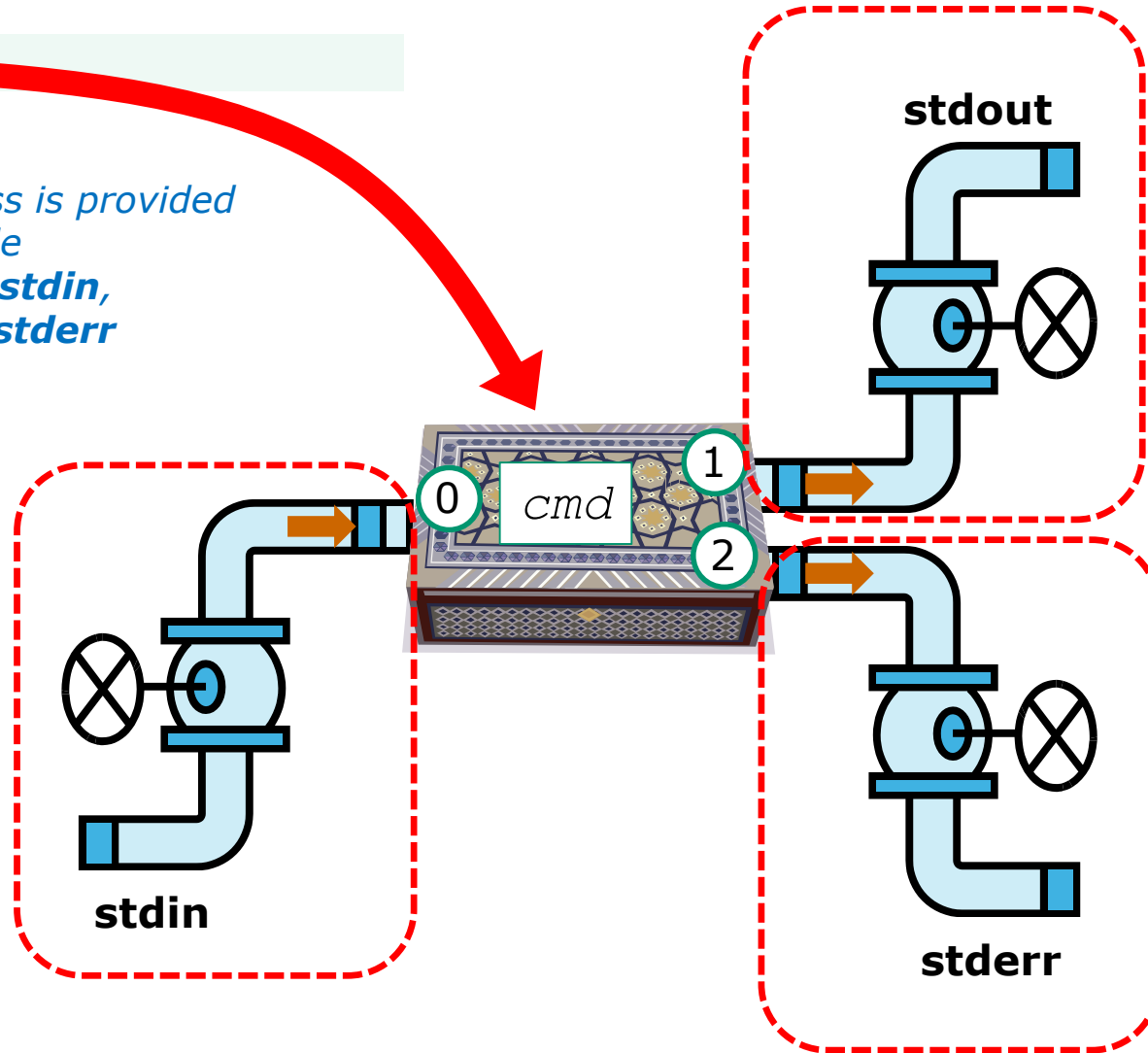
```

/home/cis90/simben $ cat
/home/cis90/simben $ cat -A letter
/home/cis90/simben $ cat < letter
/home/cis90/simben $ cat -b < letter > out
/home/cis90/simben $ cat bogus 2> /dev/null
/home/cis90/simben $ cat -e < bogus 2> /dev/null
/home/cis90/simben $ cat -e < letter > out 2> /dev/null
    
```

A program loaded into memory becomes a **process**

```
$ cmd
```

Every process is provided with three file descriptors: **stdin**, **stdout** and **stderr**

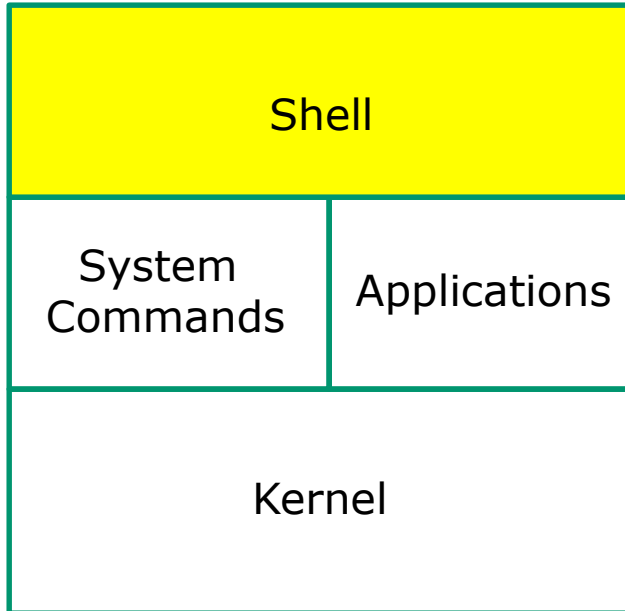




All Together Now Example



Life of the Shell



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




Shell generates the prompt string

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

The shell begins by echoing a **prompt** string to your terminal device:

- Your specific terminal device can be identified by using the **tty** command.
- The format of the prompt is defined by the contents of the PS1 variable (show with **echo \$PS1**).

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



The prompt string. In this case the PS1 variable is set to '\$PWD \$ ' which results in a prompt that shows the current location in the file tree followed by a blank, a \$, and another blank.

Activity

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

The prompt is defined by your PS1 variable

1. Look at the contents of your PS1 variable: **echo \$PS1**
2. Look at the contents of your PWD variable: **echo \$PWD**
3. Send me and yourself the contents of your prompt variable:
echo \$PS1 | mail -s "my prompt" rsimms \$LOGNAME

Paste the value of your PWD variable into the chat window when finished

User responds to the prompt by entering a command

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


Following the prompt, the user then enters a command followed by the Enter key:

- The Enter key generates a <newline> which is a "unprintable" shell metacharacter. All metacharacters have special meanings to the shell.
- The <newline> characters instructs the shell that the command line is ready to be processed.

```
/home/cis90/simben $ sort -r names > dogsinorder
```

*The user types in a command line followed by the **enter** key (on Macs the **return** key)*

Activity

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

The newline character is an invisible metacharacter that triggers the shell to process the command.

1. Put five characters in a file named *five*: **echo 12345 > five**
2. Show the file contents: **cat five**
3. Show the size of your *five* file: **ls -l five**
4. Do a hex dump of your *five* file: **xxd five** and examine the output,
hex 31 = ASCII character "1",
hex 32 = ASCII character "2" and so forth.
The 6th byte in the file is the newline character.

Put the size of your five file and the hex value of the newline character in the chat window.

*Optional: Use **man ascii** to check your answer.*

Shell parses what the user entered

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

The shell **parses** the command line entered by the user:

- The command line is carefully scanned to identify the command, options, arguments and any redirection information.
- Variables and filename expansion characters (wildcards) get processed.
- Any redirection is setup for stdin, stdout and stderr.

```
/home/cis90/simben $ sort -r names > dogsinorder
```

Parsing results: `sort` `-r` `names` `>` `dogsinorder`

The command is: **sort**

There is one option: **-r**

There is one argument: **names**

Redirection is: redirect **stdout** to a file named **dogsinorder**

Shell search for the command on the user's path

The shell now **searches** for the command on the path:

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

- The path, which is an ordered list of directories, is defined by the contents of the PATH variable. Use **echo \$PATH** to view.
- The shell will search in order each directory on the path to locate the command.
- If a command, such as xxxx, is not found, the shell will print:

-bash: xxxx: command not found

- FYI, you can search for commands on the path too, like the shell does, by using the **type** command.

```

/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 shell searches each directory on the path looking for the sort command. It finds it in the /usr/bin directory.

/usr/bin is the second directory on a CIS 90 student's path.

Activity

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

Prove to yourself that the shell will find the **sort** command in the second directory on the path, */usr/bin*.

1. Use **echo \$PATH** to view your path.
2. Starting with the first directory on the path look for the sort command:
 - Use **ls -li /usr/local/bin | grep sort**
 - Use **ls -li /usr/bin | grep sort**

Write the inode number of the sort program file in the chat window.

Example

```
$ sort -r names > dogsinorder
```

argument (points to `names`)
option (points to `-r`)
redirection (points to `>`)

The shell connects **stdout** to the **dogsinorder** file



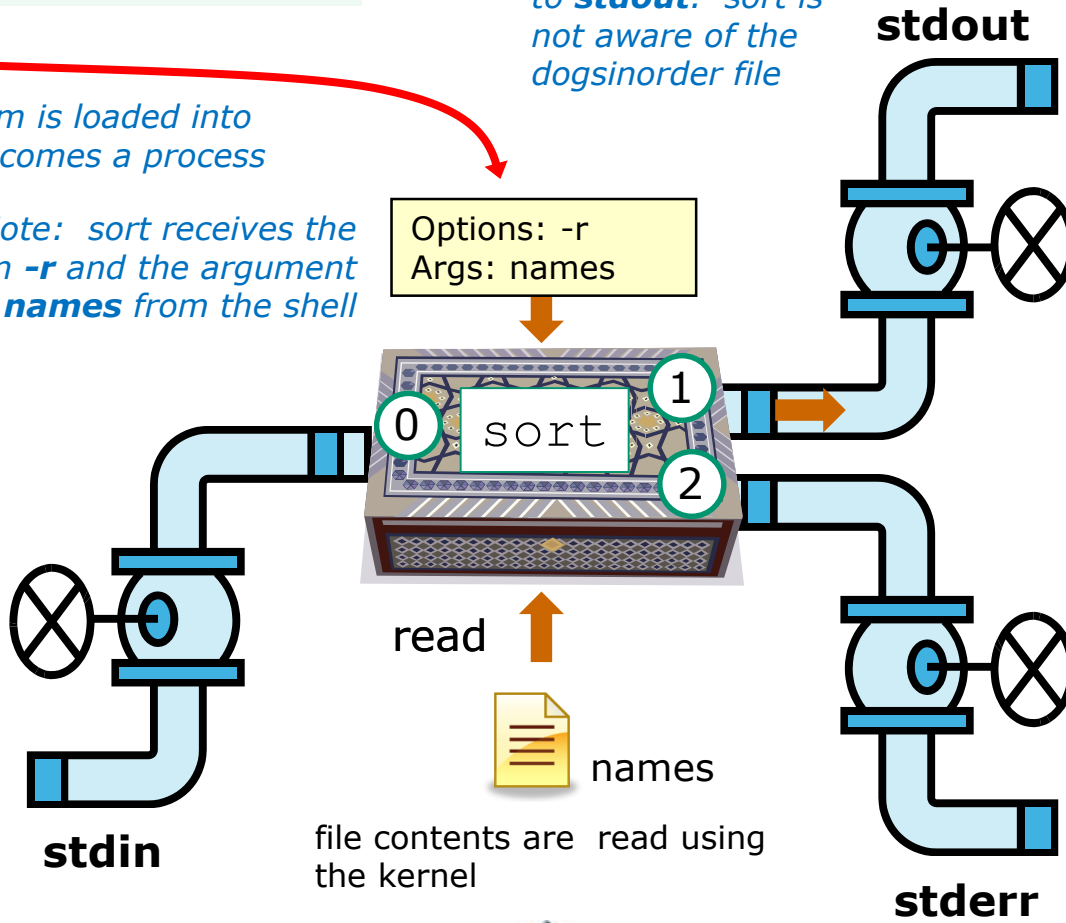
The sort program is loaded into memory and becomes a process

Note: sort receives the option **-r** and the argument **names** from the shell

Options: -r
Args: names

sort sends its output to **stdout**. sort is not aware of the dogsinorder file

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



```
benji  
duke  
homer  
star
```

file contents are read using the kernel

sort opens and reads the names file



Activity

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



```
$ sort -r names > dogsinorder
```

What two text strings parsed by the shell were handed off to the sort command to process?

Put your answer in the chat window

Example

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



While the sort process executes, the shell sleeps

Example

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



When the sort process finishes the shell wakes up and starts all over again to process the next command from the user!



Subtle Differences

What is the difference between:

head -n4 letter

and

head -n4 < letter

```
/home/cis90/simben $ head -n4 letter  
Hello Mother! Hello Father!
```

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

```
/home/cis90/simben $ head -n4 < letter  
Hello Mother! Hello Father!
```

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

head -n4 letter

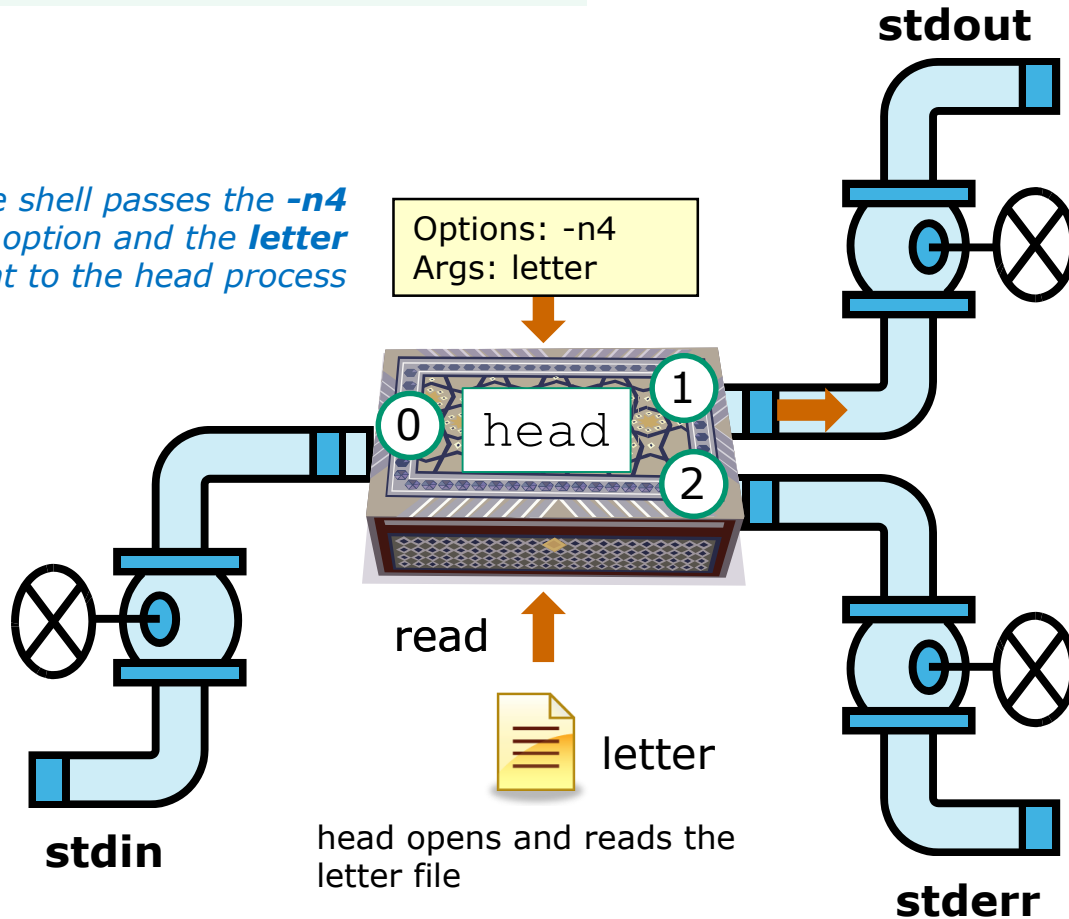
option → ← *argument*

```
$ head -n4 letter
```

```
Hello Mother! Hello Father!
```

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

The shell passes the -n4 option and the letter argument to the head process



head opens and reads the letter file

head opens and reads the letter file

head -n4 < letter

option → *redirection*

```
$ head -n4 < letter
```

stdout

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.

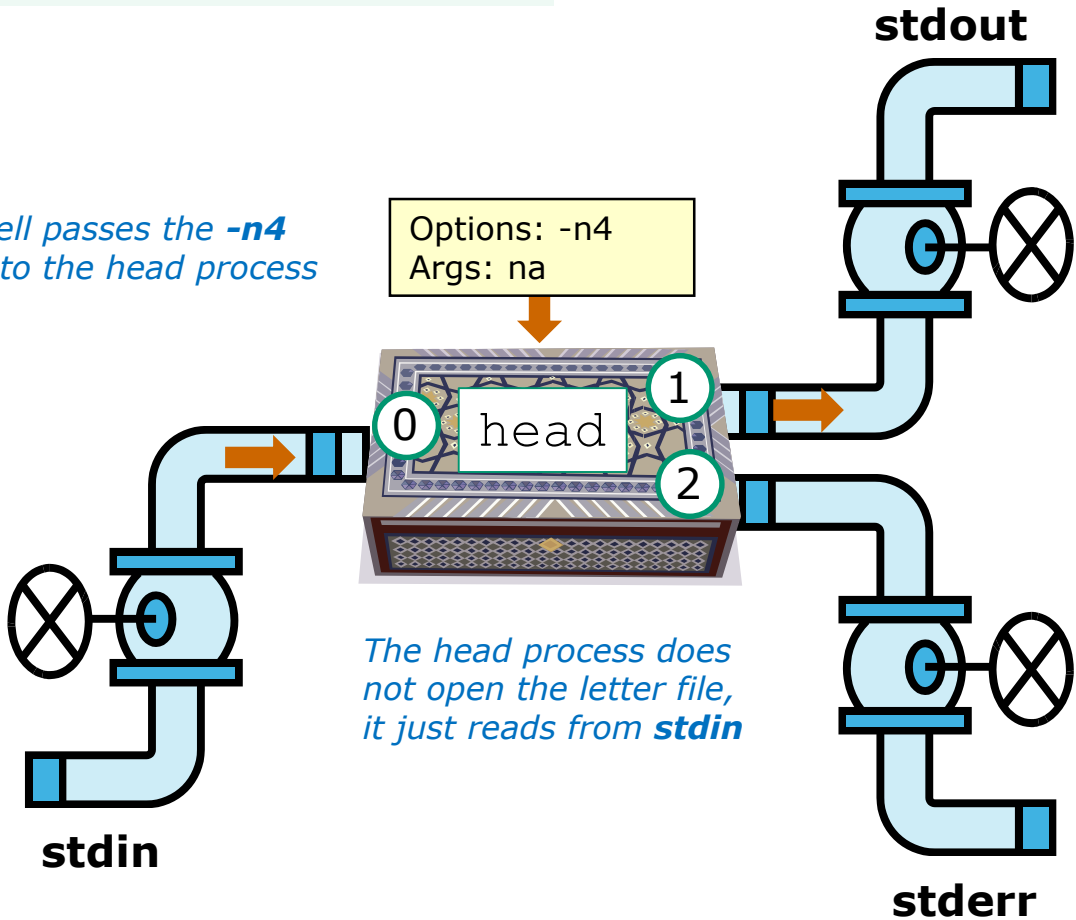
The shell passes the **-n4** option to the head process

Options: -n4
Args: na

The shell opens the letter file and connects it to **stdin**



letter



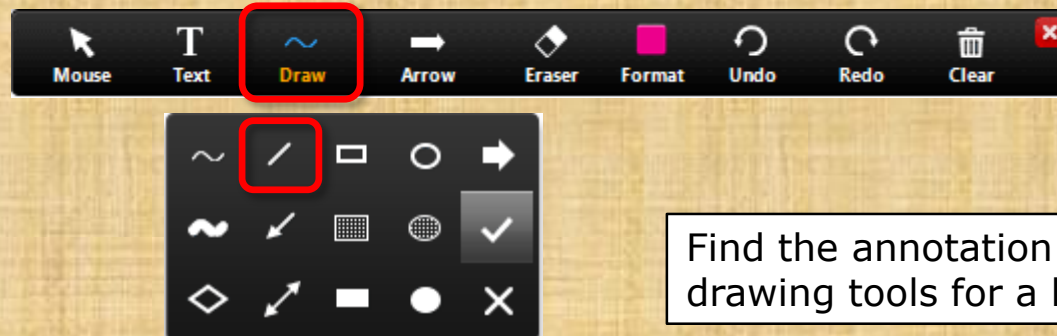
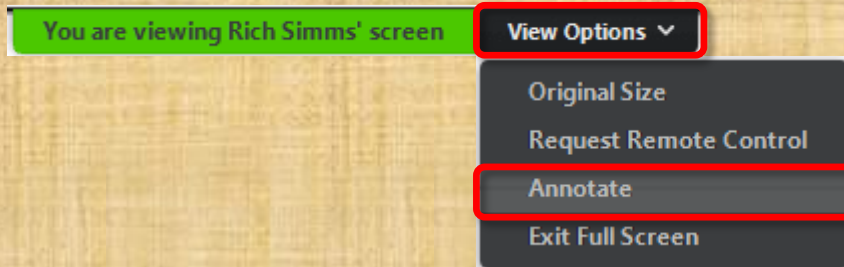
The head process does not open the letter file, it just reads from **stdin**



Errors

Instructor: Using ConferZoom annotations

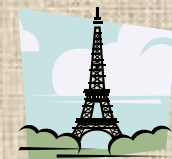
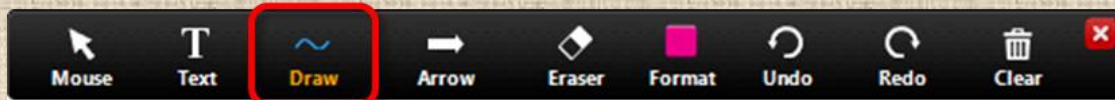
ConferZoom Annotations



Find the annotation drawing tools for a line.

View Options > Annotate Draw > "/"

CCC ConferZoom Whiteboard Activity



Select the straight line drawing tool and connect the like images

CCC Confer Whiteboard Activity

Connect with a straight line the command with the error message

Commands

\$ cat < bogus

\$ cat bogus

\$ bogus

Error messages

-bash: bogus: command not found

-bash: bogus: No such file or directory

cat: bogus: No such file or directory



CCC Confer Whiteboard Activity

Connect with a straight line the command with the error message

Commands

\$ **cat < bogus**

\$ **cat bogus**

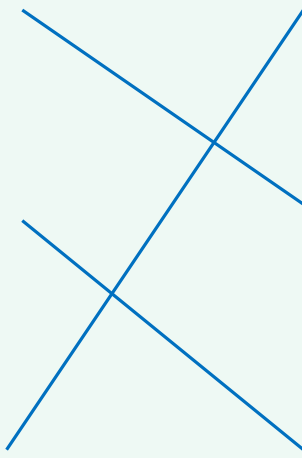
\$ **bogus**

Error messages

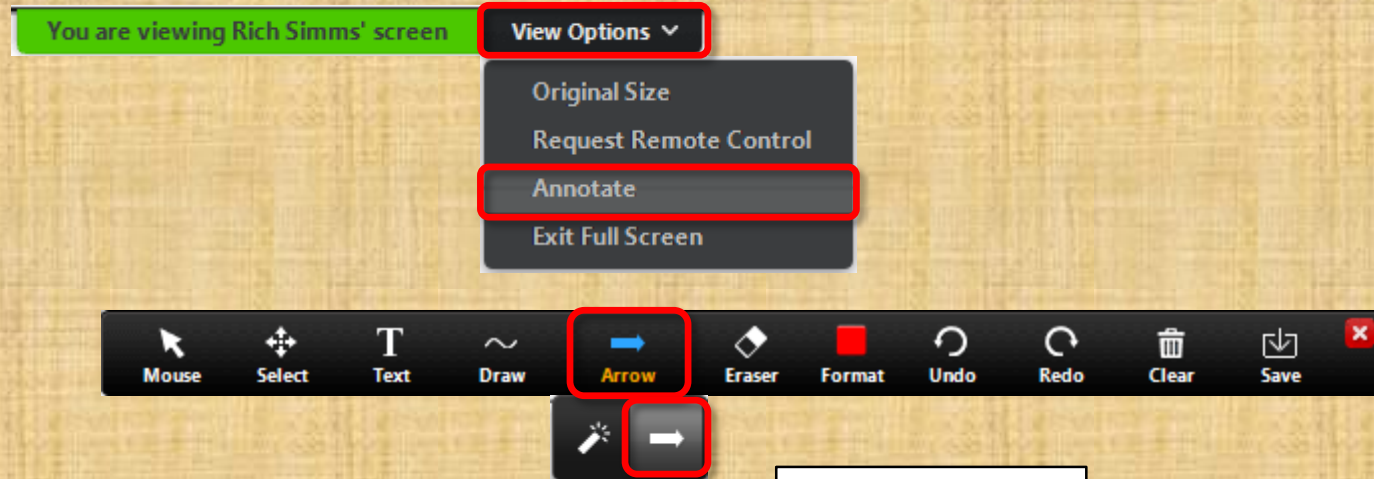
-bash: bogus: command not found

-bash: bogus: No such file or directory

cat: bogus: No such file or directory



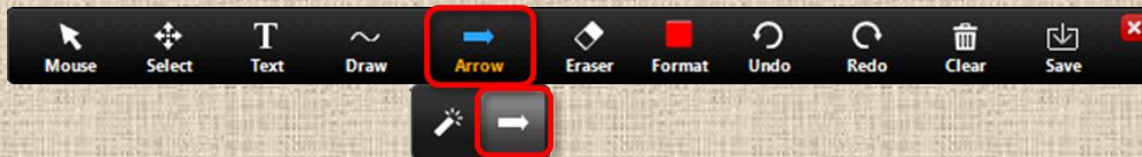
ConferZoom Annotations



Find the
arrow
annotation

View Options > Annotate Draw > Spotlight or Arrow > Arrow

CCC Confer Whiteboard Activity



Point (don't click) at the number called out by the instructor with your arrow annotation.

1

2

3

4

CCC ConferZoom Whiteboard Activity

Shell Steps

1) Prompt

2) Parse

3) Search

4) Execute

5) Nap

6) Repeat

Given: There is no file named *bogus*

```
[rsimms@oslab ~]$ cat bogus  
cat: bogus: No such file or directory
```

Point to the shell step where the error message was generated

CCC ConferZoom Whiteboard Activity

Given: There is no file named *bogus*

```
[rsimms@oslab ~]$ bogus  
-bash: bogus: command not found
```

Point to the shell step where the error message was generated

Shell Steps

1) Prompt

2) Parse

3) Search

4) Execute

5) Nap

6) Repeat

CCC ConferZoom Whiteboard Activity

Given: There is no file named *bogus*

```
[rsimms@oslab ~]$ cat < bogus  
-bash: bogus: No such file or directory
```

Point to the shell step where the error message was generated

Shell Steps

1) Prompt

2) Parse

3) Search

4) Execute

5) Nap

6) Repeat

CCC ConferZoom Whiteboard Activity

Given: There is no file named *bogus*

```
[rsimms@oslab ~]$ bogus1 < bogus2  
-bash: bogus2: No such file or directory
```

Point to the shell step where the error message was generated

Shell Steps

1) Prompt

2) Parse

3) Search

4) Execute

5) Nap

6) Repeat

CCC ConferZoom Whiteboard Activity

Given: There is no file named *bogus*

```
[rsimms@oslab ~]$ cat bogus  
cat: bogus: No such file or directory 1) Execute
```

```
[rsimms@oslab ~]$ bogus  
-bash: bogus: command not found 3) Search
```

```
[rsimms@oslab ~]$ cat < bogus  
-bash: bogus: No such file or directory 2) Parse
```

```
[rsimms@oslab ~]$ bogus1 < bogus2  
-bash: bogus2: No such file or directory 2) Parse
```

2 > & 1

FYI





It's descriptor clobbering time!

```
/home/cis90/simben $ bc > calculations 2> calculations  
2+2  
7/0  
3+3  
quit
```

```
/home/cis90/simben $ cat calculations  
Ru6  
ime error (func=(main), adr=5): Divide by zero
```

*Oops! Its not a good idea to redirect **stdout** and **sderr** to the same file because they clobber each other!*



It's descriptor clobbering time!

```
/home/cis90/simben $ bc > calculations 2> calculations
```

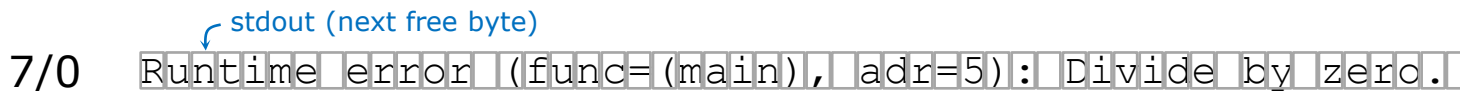


stderr (next free byte)

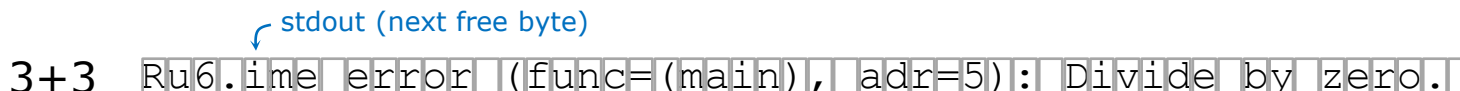


stderr (next free byte)

The <newline> character is represented by a "."



stderr (next free byte)



stderr (next free byte)

```
/home/cis90/simben $ cat calculations
```

```
Ru6
```

```
ime error (func=(main), adr=5): Divide by zero
```

Each file descriptor keeps its own separate index into the calculations file for where to write the next line.



It's descriptor collaboration time!

```
/home/cis90/simben $ bc > calculations 2>&1  
2+2  
7/0  
3+3  
quit
```

```
/home/cis90/simben $ cat calculations  
4  
Runtime error (func=(main), adr=5): Divide by zero  
6
```

*stdout is redirected
to calculations and
stderr is redirected
to the same file
attached to stdout*

*This is the correct way to redirect **stdout** and **stderr** to
the same file*

More on I/O

(input/output)

programming examples





C Program I/O example Execute the program

Running the simple program.

```
What is your name stranger? Benji  
Well I'm very pleased to meet you, Benji
```

```
/home/cis90/simben $ simple > greet  
What is your name stranger? Homer  
/home/cis90/simben $ cat greet  
Well I'm very pleased to meet you, Homer
```

```
/home/cis90/simben $ echo Duke | simple > greet  
What is your name stranger? /home/cis90/simben $ cat greet  
Well I'm very pleased to meet you, Duke
```



C Program I/O example View the program

```
/home/cis90/simben/bin $ cat simple.c
char question[] = "What is your name stranger? ";
char greeting[] = "Well I'm very pleased to meet you, ";
char buffer[80];
main()
{
    int len;

    write(2, question, sizeof(question));
    len = read(0, buffer, 80);
    write(1, greeting, sizeof(greeting));
    write(1, buffer, len);
}
```

*What do you think the reads and writes to
0, 1 and 2 mean in the code above?*



C Program I/O example View the program

```

/home/cis90/simben/bin $ cat simple.c
char question[] = "What is your name stranger? ";
char greeting[] = "Well I'm very pleased to meet you, ";
char buffer[80];
main()
{
    int len;

    write(2, question, sizeof(question));
    len = read(0, buffer, 80);
    write(1, greeting, sizeof(greeting));
    write(1, buffer, len);
}

```

This simple program asks for a name, then responds with a greeting using the name



C Program I/O example

Compile the program

The make command is used to compile a C source text file into a binary executable

```
/home/cis90/simben/bin $ make simple  
cc      simple.c      -o simple
```

Unlike a bash script, the C program source code must be compiled into a binary executable before it can be run



C Program I/O example Execute the program

```
/home/cis90/simben/bin $ simple  
What is your name stranger? Rich  
Well I'm very pleased to meet you, Rich
```

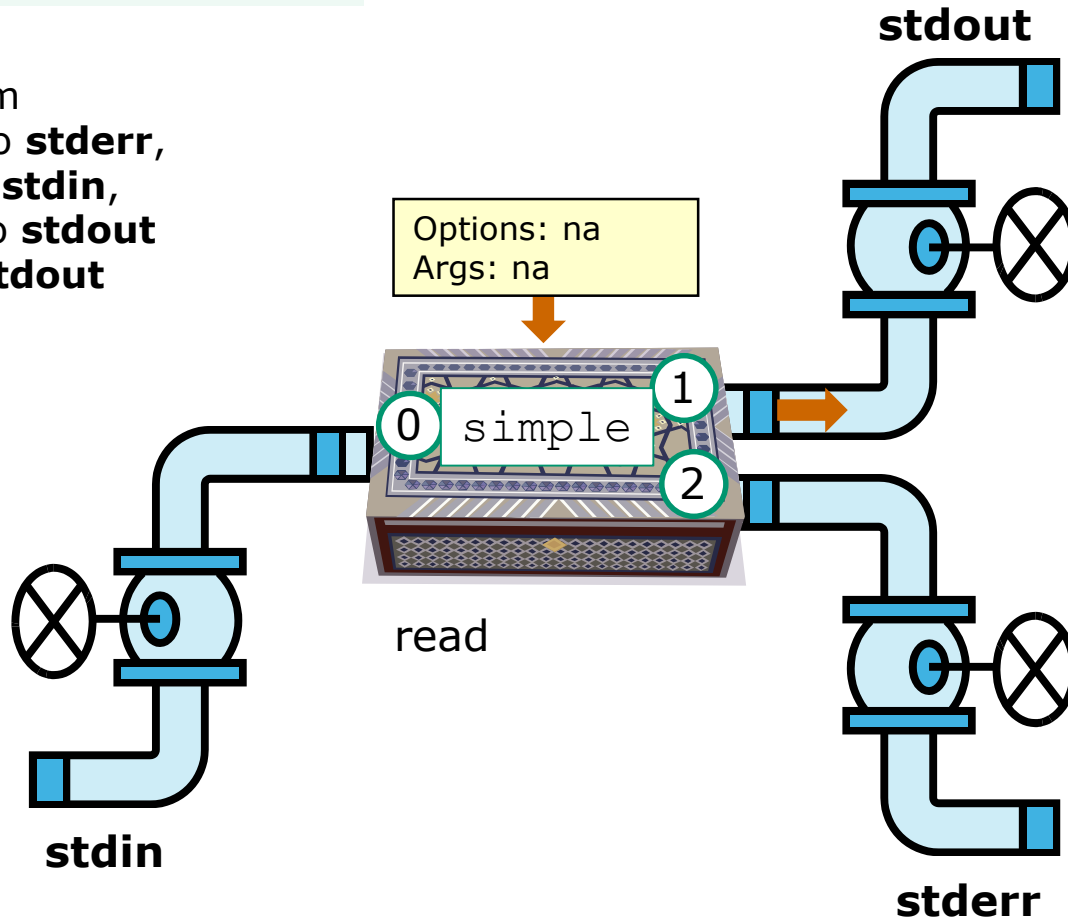
Running the simple program.

C Program I/O example

```
$ simple
```

The **simple** program

1. writes question to **stderr**,
2. reads input from **stdin**,
3. writes greeting to **stdout**
4. writes name to **stdout**



2

Rich

3

Well I'm very
pleased to meet
you, Rich

4

1

What is your name
stranger?



C Program I/O example

```
/home/cis90/simben/bin $ simple > myfile  
What is your name stranger? Rich
```

```
/home/cis90/simben/bin $ cat myfile  
Well I'm very pleased to meet you, Rich
```

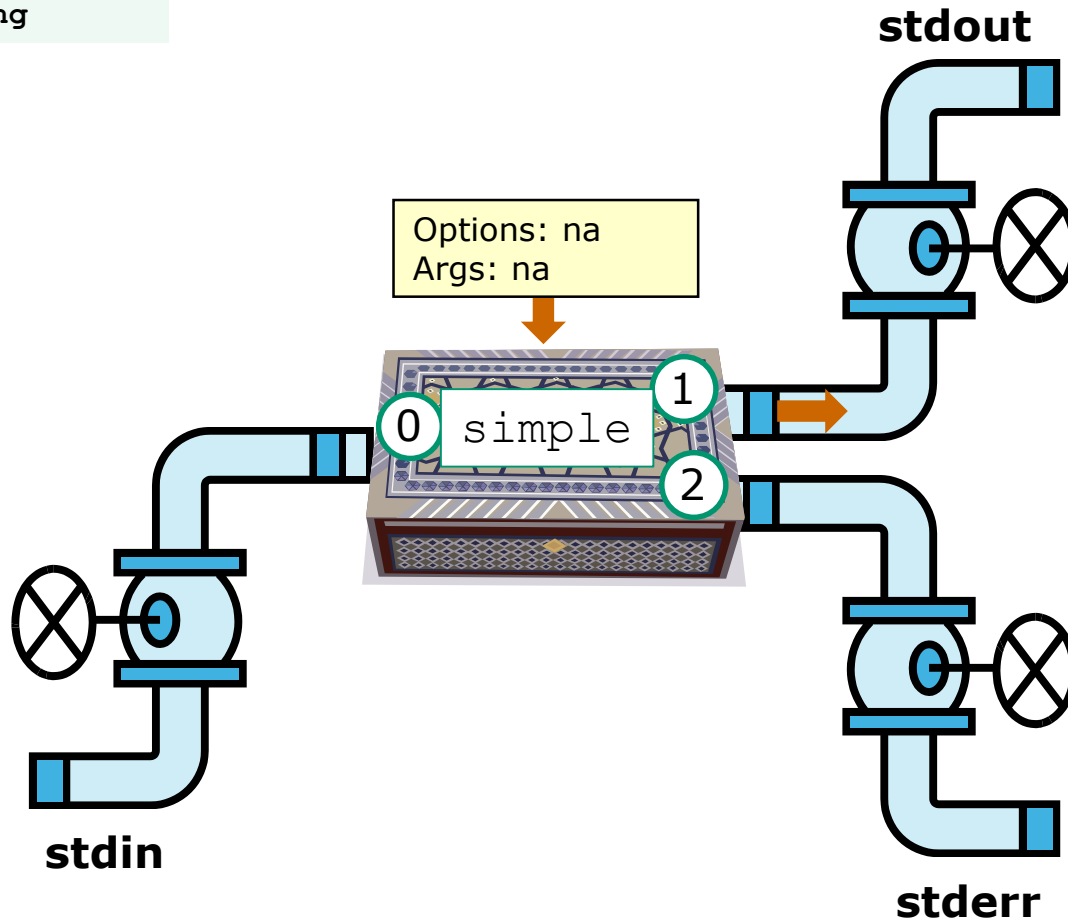
*In this example,
output has been
redirected to a file
named myfile.*

*The simple program has no special knowledge (coding instructions) for a file named myfile. It just writes to **stdout** and that output will go to wherever **stdout** had been directed.*

C Program I/O example

```
$ simple > greeting
```

redirection



greeting

```
Well I'm very  
pleased to meet  
you, Rich
```

```
Rich
```

```
What is your name  
stranger?
```

Activity

1. Change to your bin directory
cd ~/bin
2. Copy the C source code from the depot directory
cp ../../depot/simple.c .
3. Look at your program
cat simple.c
4. Compile the program
make simple
5. Run the program
simple

C++ Program I/O example View the program

FYI
only

```

/home/cis90/simben/bin $ cat simpleplus.cpp
#include <iostream>
using namespace std;

int main() {
    string question = "What is your name stranger? ";
    cerr << question;
    string buffer;
    cin >> buffer;
    string greeting = "Well I'm very pleased to meet you, ";
    cout << greeting << buffer << endl;
    return 0;
}

```

*Write question to **stderr***

*Read name from **stdin***

*Write greeting and name to **stdout***

This program is available in the depot directory

C++ Program I/O example

Compile the program

FYI
only

The make command is used to compile a C++ source text file into a binary executable

```
/home/cis90/simben/bin $ make simpleplus  
g++      simpleplus.cpp  -o simpleplus
```

Unlike a bash script, the C++ program source code must be compiled into a binary executable before it can be run

C++ Program I/O example Execute the program



```
/home/cis90/simben/bin $ simpleplus  
What is your name stranger? Rich  
Well I'm very pleased to meet you, Rich
```

Running the simpleplus program



Activity

1. Change to your bin directory
cd ~/bin
2. Copy the C++ source code from the depot directory
cp ../../depot/simpleplus.cpp .
3. Look at your program
cat simpleplus.cpp
4. Compile the program
make simpleplus
5. Run the program
simpleplus

Python Script I/O example

View the program

FYI
only

```
/home/cis90/simben $ cat simple.py  
import sys  
sys.stderr.write("What is your name stranger? ") Output question to stderr  
name = sys.stdin.readline() Input name from stdin  
sys.stdout.write("Well I'm very pleased to meet you, " + name)
```

 *Output greeting and name to **stdout***

This program is available in the depot directory

Python Script I/O example View the program

FYI
only

```
/home/cis90/simben $ python simple.py  
What is your name stranger? Rich  
Well I'm very pleased to meet you, Rich  
/home/cis90/simben $
```

Running the python simple.py script



Activity

1. Change to your bin directory
cd ~/bin
2. Copy the python script from the depot directory
cp ../../depot/simple.py .
3. Look at your program
cat simple.py
4. Run the script
python simple.py

umask

(review)

Review - applying umask bits

Example umask setting

```
/home/cis90/simben/lesson9 $ umask  
0002
```

this mask indicates which permissions should NOT be set on new files or directories

New file - start with 666 and apply mask

666	110	110	110
002	000	000	010
	↓	↓	↓
	↓	↓	↓
	↓	↓	↓
664	110	110	100

```
/home/cis90/simben/lesson9 $ touch newfile  
/home/cis90/simben/lesson9 $ ls -l newfile  
-rw-rw-r-- 1 simben cis90 0 Oct 27 07:22 newfile
```

New directory - start with 777 and apply mask

777	111	111	111
002	000	000	010
	↓	↓	↓
	↓	↓	↓
	↓	↓	↓
775	111	111	101

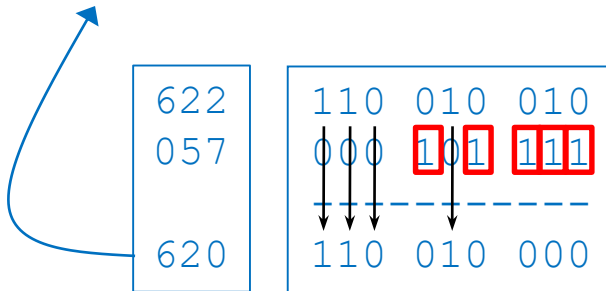
```
/home/cis90/simben/lesson9 $ mkdir newdir  
/home/cis90/simben/lesson9 $ ls -ld newdir  
drwxrwxr-x 2 simben cis90 4096 Oct 27 07:23 newdir
```

Any umask bits set to 1 removes the corresponding permission bit for future new files and directories

Review - Copying files

```
/home/cis90/simben $ umask 057 Example umask setting
/home/cis90/simben $ umask
0057
```

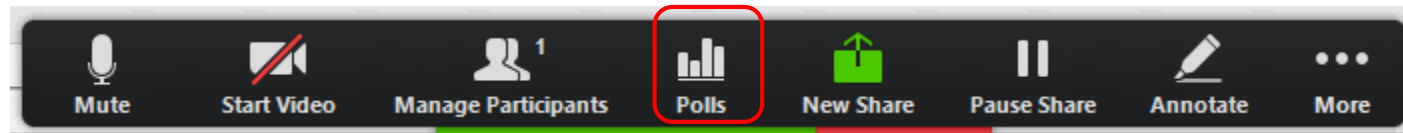
```
/home/cis90/simben $ chmod 622 myfile
/home/cis90/simben $ cp myfile myfile.bak
/home/cis90/simben $ ls -l myfile*
-rw--w--w-. 1 simben90 cis90 0 Mar 24 17:50 myfile
-rw--w----. 1 simben90 cis90 0 Mar 24 17:51 myfile.bak
```



Copied file - start with original file's permissions and apply the mask

*Remember, for new files resulting from copying, instead of using the **default permissions** (666 for file and 777 for directory), use the **original file permissions** as the starting point for the mask to be applied to.*

Rich's CCC ConferZoom poll setup



Polls

Polling 1: A-D single choice Edit

1. Select the best answer (Multiple choice)

A

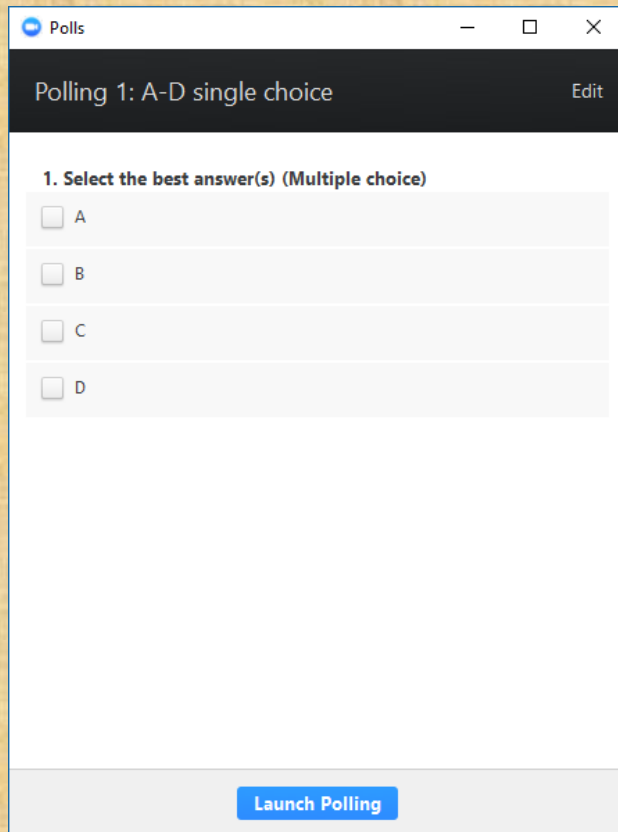
B

C

D

[Launch Polling](#)

Activity



The screenshot shows a web browser window with the title "Polls". The main content area is titled "Polling 1: A-D single choice" and includes an "Edit" link. Below the title, there is a question: "1. Select the best answer(s) (Multiple choice)". There are four radio button options labeled A, B, C, and D. At the bottom of the form, there is a blue button labeled "Launch Polling".

Which pizza is the best?

- A. Round Table
- B. Pizza My Heart
- C. Tony & Alba's
- D. Upper Crust

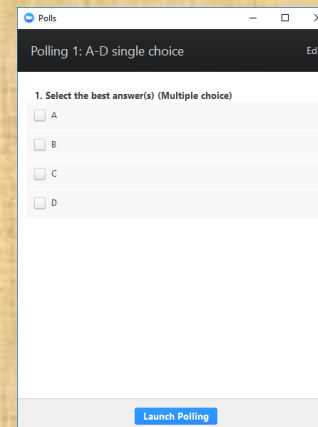
Respond to the poll above

Activity

I want to change the permissions on an existing file

Which command does this?

- A) stat
- B) ls -l
- C) chmod
- D) umask



The screenshot shows a web browser window titled "Polls". The main content area displays "Polling 1: A-D single choice" with an "Edit" link. Below this, it says "1. Select the best answer(s) (Multiple choice)". There are four radio button options: A, B, C, and D. At the bottom of the poll form, there is a blue button labeled "Launch Polling".

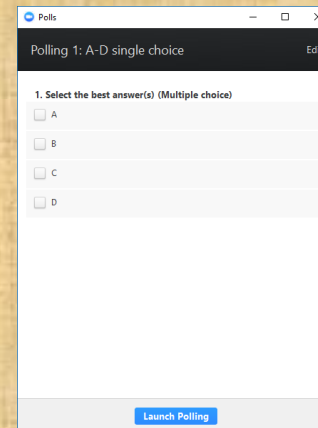
Respond to the poll above

Activity

I want to restrict specific permissions on files that have not been created yet

Which command does this?

- A) stat
- B) ls -l
- C) chmod
- D) umask



The screenshot shows a web-based poll interface. At the top, it says "Polls" and "Polling 1: A-D single choice" with an "Edit" link. Below that, it asks "1. Select the best answer(s) (Multiple choice)". There are four radio button options: A, B, C, and D. At the bottom, there is a blue button labeled "Launch Polling".

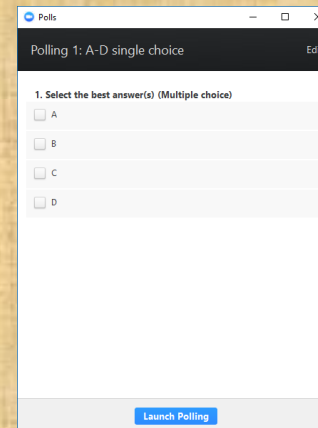
Respond to the poll above

Activity

I want to show the owner of a file and its permissions in mnemonic format e.g. `rwxr-xr-x`

Which command does this?

- A) `stat`
- B) `ls -l`
- C) `chmod`
- D) `umask`



The screenshot shows a web-based poll interface. At the top, it says "Polls" and "Polling 1: A-D single choice" with an "Edit" link. Below that, the question is "1. Select the best answer(s) (Multiple choice)". There are four radio button options: A, B, C, and D. At the bottom, there is a "Launch Polling" button.

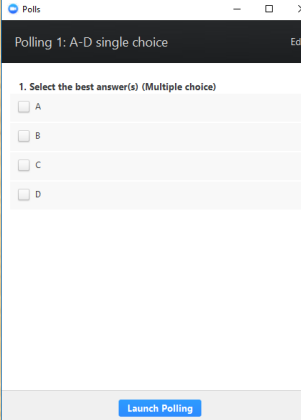
Respond to the poll above

Activity

**I want to show the permissions on a file in numeric format
e.g. 750**

Which command does this?

- A) stat
- B) ls -l
- C) chmod
- D) umask



The screenshot shows a web-based poll interface. At the top, it says "Polls" and "Polling 1: A-D single choice" with an "Edit" link. Below that, it asks "1. Select the best answer(s) (Multiple choice)". There are four radio button options: A, B, C, and D. At the bottom, there is a "Launch Polling" button.

Respond to the poll above



More Pipeline Practice

Pipelines

Task

Record the last times Benji Simms (in CIS 90) logged in on a Monday to a file named *mylog* AND count them.

grep Benji /etc/passwd

```
simben76:x:1501:1076:Benji Simms:/home/cis76/simben:/bin/bash
simben90:x:1201:1090:Benji Simms:/home/cis90/simben:/bin/bash
```

last

```
last | grep simben90
```

```
last | grep simben90 | grep "Mon"
```

```
last | grep simben90 | grep "Mon" | tee mylog
```

```
cat mylog
```

```
last | grep simben90 | grep "Mon" | tee mylog | wc -l
```

```
cat mylog
```

Class Exercise Pipeline Tasks

Task

Count the last times Rich Simms was logged in on a Tuesday and record them in a file named mylog

```
grep "?????" /etc/passwd
```

```
last | grep ??????
```

```
last | grep ?????? | grep "Tue"
```

```
last | grep ?????? | grep "Tue" | ??? mylog  
cat mylog
```

```
last | grep ?????? | grep "Tue" | ??? mylog | wc -?  
cat mylog
```

Put your answer in the chat window.

Pipelines

Task

Print your last name as shown in /etc/passwd:

```
cat /etc/passwd
```

```
cat /etc/passwd | grep $LOGNAME
```

```
cat /etc/passwd | grep $LOGNAME | cut -f 5 -d ":"
```

```
cat /etc/passwd | grep $LOGNAME | cut -f 5 -d ":" | cut -f2 -d" "
```

Class Exercise Pipeline Tasks

Task

What is the first name of the user milhom90?

```
cat /etc/passwd
```

```
cat /etc/passwd | grep ????????
```

```
cat /etc/passwd | grep ????????
```

```
cat /etc/passwd | grep ????????
```

Put your answer in the chat window.

Pipelines

Task

Print a sorted list of the first names for all CIS 76 students

```
cat /etc/passwd
```

```
cat /etc/passwd | grep cis76
```

```
cat /etc/passwd | grep cis76 | cut -f 5 -d ":"
```

```
cat /etc/passwd | grep cis76 | cut -f 5 -d ":" | cut -f1 -d" "
```

```
cat /etc/passwd | grep cis76 | cut -f 5 -d ":" | cut -f1 -d" " | sort
```


Class Exercise Pipeline Tasks

Task

Print a sorted list of the first names for CIS 90 students

```
cat /etc/??????
```

```
cat /etc/?????? | grep cis??
```

```
cat /etc/?????? | grep ????? | cut -f ? -d "?"
```

```
cat /etc/?????? | grep ????? | cut -f ? -d "?" | cut -f? -d"?" | ????
```

Put your list in the chat window.



More Review for Test 2

Review Activity



1) There is a file that contains all the user accounts on a system. The name of this file is *passwd* and it is found in the */etc* directory. Cat this file and look at it. What is the ABSOLUTE pathname of this file?

Put your answer in the chat window

Review Activity

2) In what directory does the **lspci** command reside?

Put your answer in the chat window

Review Activity

3) From your Neruda directory, what is the relative pathname to the directory containing the **lspci** command?

Put your answer in the chat window

```
/home/cis90/simben $ tree ~milhom90/trouble/  
/home/cis90/milhom/trouble/
```

```
├── food
│   ├── pies
│   │   ├── apple
│   │   ├── blueberry
│   │   ├── cherry
│   │   └── mud
│   └── sandwiches
│       ├── grilled_cheese
│       ├── gyro
│       └── polish-dog
├── keep
│   └── going
│       └── down
│           └── till
│               └── the
│                   └── bottom
│                       └── names
└── rocks
    ├── igneous
    │   ├── basalt
    │   ├── diorite
    │   ├── granite
    │   └── pumice
    └── sedimentary
        ├── limestone
        ├── sandstone
        └── shale
```

```
12 directories, 15 files  
/home/cis90/simben $
```

Tree diagram of Homer's trouble directory

```
tree ~milhom90/trouble/
```

To make your own trouble directory use:
cd
tar xvzf ~/../depot/trouble.tgz

Permissions on all files in Homer's trouble directory

```

/home/cis90/simben $ find ~milhom90/trouble/ -exec ls -ld {} \;
drwxr-xr-x. 5 milhom90 cis90 43 Oct 23 13:54 /home/cis90/milhom/trouble/
drwxr-xr-x. 4 milhom90 cis90 36 Oct 23 13:53 /home/cis90/milhom/trouble/food
drwxr-xr-x. 2 milhom90 cis90 61 Oct 23 13:37 /home/cis90/milhom/trouble/food/pies
-rw-r--r--. 1 milhom90 cis90 17 Oct 23 13:38 /home/cis90/milhom/trouble/food/pies/apple
-rw-r--r--. 1 milhom90 cis90 18 Oct 23 13:38 /home/cis90/milhom/trouble/food/pies/cherry
-rw-r--r--. 1 milhom90 cis90 21 Oct 23 13:38 /home/cis90/milhom/trouble/food/pies/blueberry
-rw-r--r--. 1 milhom90 cis90 15 Oct 23 13:38 /home/cis90/milhom/trouble/food/pies/mud
drw-r-xr-x. 2 milhom90 cis90 58 Oct 23 13:41 /home/cis90/milhom/trouble/food/sandwiches
-rw-r--r--. 1 milhom90 cis90 22 Oct 23 13:41 /home/cis90/milhom/trouble/food/sandwiches/grilled_cheese
-rw-r--r--. 1 milhom90 cis90 13 Oct 23 13:42 /home/cis90/milhom/trouble/food/sandwiches/gyro
-rw-r--r--. 1 milhom90 cis90 19 Oct 23 13:42 /home/cis90/milhom/trouble/food/sandwiches/polish-dog
drwxr-xr-x. 4 milhom90 cis90 40 Oct 23 13:43 /home/cis90/milhom/trouble/rocks
dr-xr-xr-x. 2 milhom90 cis90 53 Oct 23 13:45 /home/cis90/milhom/trouble/rocks/sedimentary
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:45 /home/cis90/milhom/trouble/rocks/sedimentary/sandstone
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:45 /home/cis90/milhom/trouble/rocks/sedimentary/shale
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:45 /home/cis90/milhom/trouble/rocks/sedimentary/limestone
drwxr-xr-x. 2 milhom90 cis90 64 Oct 23 13:48 /home/cis90/milhom/trouble/rocks/igneous
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:48 /home/cis90/milhom/trouble/rocks/igneous/granite
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:48 /home/cis90/milhom/trouble/rocks/igneous/diorite
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:48 /home/cis90/milhom/trouble/rocks/igneous/basalt
-rw-r--r--. 1 milhom90 cis90 0 Oct 23 13:48 /home/cis90/milhom/trouble/rocks/igneous/pumice
drwxr-xr-x. 3 milhom90 cis90 19 Oct 23 13:54 /home/cis90/milhom/trouble/keep
drwxr-xr-x. 3 milhom90 cis90 18 Oct 23 13:54 /home/cis90/milhom/trouble/keep/going
drwxr-xr-x. 3 milhom90 cis90 18 Oct 23 13:54 /home/cis90/milhom/trouble/keep/going/down
drwxr-xr-x. 3 milhom90 cis90 17 Oct 23 13:54 /home/cis90/milhom/trouble/keep/going/down/till
drwxr-xr-x. 3 milhom90 cis90 20 Oct 23 13:54 /home/cis90/milhom/trouble/keep/going/down/till/the
drwxr-xr-x. 2 milhom90 cis90 19 Oct 23 13:55 /home/cis90/milhom/trouble/keep/going/down/till/the/bottom
-rw-r--r--. 1 milhom90 cis90 27 Oct 23 13:55 /home/cis90/milhom/trouble/keep/going/down/till/the/bottom/names
/home/cis90/simben $

```

```
find ~milhom90/trouble/ -exec ls -ld {} \;
```

Review Activity

Homer

```

/home/cis90/milhom $ ls -l trouble/food/sandwiches/
ls: cannot access trouble/food/sandwiches/grilled_cheese: Permission denied
ls: cannot access trouble/food/sandwiches/gyro: Permission denied
ls: cannot access trouble/food/sandwiches/polish-dog: Permission denied
total 0
-?????????? ? ? ? ? ?      ? grilled_cheese
-?????????? ? ? ? ? ?      ? gyro
-?????????? ? ? ? ? ?      ? polish-dog
    
```

Benji

```

/home/cis90/simben $ ls -l ~milhom90/trouble/food/sandwiches/
total 12
-rw-r--r--. 1 milhom90 cis90 22 Oct 23 13:41 grilled_cheese
-rw-r--r--. 1 milhom90 cis90 13 Oct 23 13:42 gyro
-rw-r--r--. 1 milhom90 cis90 19 Oct 23 13:42 polish-dog
    
```

4) Homer is getting a weird long listing on his *sandwiches* directory. Benji isn't. How can Homer fix this?

Put your answer in the chat window

Review Activity

Homer

```
/home/cis90/milhom/trouble/rocks/sedimentary $ touch siltstone  
touch: cannot touch `siltstone': Permission denied  
/home/cis90/milhom/trouble/rocks/sedimentary $
```

5) Homer can't add any more rock files to his *sedimentary* directory? How can he fix this?

Put your answer in the chat window

Review Activity Answers

1) /etc/passwd

2) /usr/sbin

3) ../../../../usr/bin

4) Homer does not have execute permission for user on his sandwiches directory:

```
/home/cis90/milhom $ ls -ld trouble/food/sandwiches/  
drw-r-xr-x. 2 milhom90 cis90 58 Oct 23 13:41 trouble/food/sandwiches/
```

Homer can fix using: **chmod u+x trouble/food/sandwiches/**

5) Homer does not have write permission for user on his sedimentary directory:

```
/home/cis90/milhom $ ls -ld trouble/rocks/sedimentary/  
dr-xr-xr-x. 2 milhom90 cis90 53 Oct 23 13:45 trouble/rocks/sedimentary/
```

Homer can fix using: **chmod u+w trouble/rocks/sedimentary/**

More on pipelines

Not all commands are filters (filters read from stdin and write to stdout)

*The **wc** command is a filter.*

```
/home/cis90/simben $ head -n2 poems/Anon/nursery
Jack and Jill went up the hill
to fetch a pail of water.
/home/cis90/simben $ head -n2 poems/Anon/nursery | wc -l
2
/home/cis90/simben $
```

*But the **echo** command isn't (doesn't read from **stdin**)*

```
/home/cis90/simben $ head -n2 poems/Anon/nursery | echo
Oops .... this doesn't work!
/home/cis90/simben $
```

xargs command

xargs to the rescue!



```
/home/cis90/simben $ head -n2 poems/Anon/nursery | xargs echo  
Jack and Jill went up the hill to fetch a pail of water.
```

The **xargs** command will read **stdin** and call another command using the input as the arguments.

Another example

Why can't Benji make a banner using the output of the date command?

```
/home/cis90/simben $ date | banner  
Enter a string of up to 10 characters.  
/home/cis90/simben $
```

huh? Oh, this is what banner prints when it receives no arguments on the command line

Because banner is not a filter and does not read from stdin!

Another example

```
/home/cis90/simben $ date | xargs banner
```

```
# # ##### # #  
## ## # ## #  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
# # ##### # #
```

```
##### #####  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
##### #####
```

```
##### #####  
# # # # # # # #  
# # # # # # # #  
##### #####  
# # # # # # # #  
# # # # # # # #  
##### #####
```

```
# # ##### # #  
## ## # ## #  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
# # # # # # # #  
##### #####
```

```
##### #####  
# # # # # # # #  
# # # # # # # #  
##### # # # #  
# # # # # # # #  
# # # # # # # #  
##### # # # #
```

```
##### # # # #  
# # # # # # # #  
# # # # # # # #  
##### # # # #  
# # # # # # # #  
# # # # # # # #  
##### # # # #
```

xargs to the rescue again!

Not all commands are filters (filters read from stdin and write to stdout)

*The **ls** command does not read from **stdin** either*

```
/home/cis90/simben $ find poems -type d  
poems  
poems/Shakespeare  
poems/Yeats  
poems/Anon  
poems/Blake
```

```
/home/cis90/simben $ find poems -type d | ls -ld  
drwxr-xr-x. 18 simben90 cis90 4096 Oct 22 09:49 .  
/home/cis90/simben $
```

Benji was hoping that he could get a long listing of his poems directory and all its sub-directories. Instead he gets a long listing of his home directory!

Not all commands are filters (filters read from stdin and write to stdout)

xargs to the rescue again!

```

/home/cis90/simben $ find poems -type d | xargs ls -ld
drwxr-xr-x. 6 simben90 cis90 4096 Oct 20 15:06 poems
drwxr-xr-x. 2 simben90 cis90 4096 Oct  5 10:26 poems/Anon
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Blake
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Shakespeare
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Yeats
/home/cis90/simben $
  
```

*The **ls** command is not a filter so it does not read from **stdin***

***xargs** reads the names of the files found by the **find** command and uses them as arguments on the **ls -ld** command*

Not all commands are filters (filters read from stdin and write to stdout)

```
/home/cis90/simben $ find poems -type d -exec ls -ld {} \;
drwxr-xr-x. 6 simben90 cis90 4096 Oct 20 15:06 poems
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Shakespeare
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Yeats
drwxr-xr-x. 2 simben90 cis90 4096 Oct 5 10:26 poems/Anon
drwxr-xr-x. 2 simben90 cis90 4096 Oct 20 15:06 poems/Blake
/home/cis90/simben $
```

*The **find** command also has a **-exec** option that will run a command on what is found. The **{}** represent the arguments which are names of files found by the **find** command.*



Things that Hide

Memorize This

How to locate files:

To locate by name ... use **find**

To locate by user, type, group, etc¹. ... use **find**

To locate by contents² within specific files ... use **grep**

To locate by contents² in a branch of the file tree ... use recursive **grep**

1 File inode properties (e.g. shown in a long listing)

2 File data (e.g. shown when catting a file)

Finding Things

Task

Find all files in the `/usr/share/doc` branch of the file tree that are named "BUGS"

```
find /usr/share/doc -name "BUGS"
```

```
/usr/share/doc/ppl-0.10.2/BUGS  
/usr/share/doc/ltrace-0.5/BUGS  
/usr/share/doc/perl-IO-Socket-SSL-1.31/BUGS  
/usr/share/doc/glibc-2.12/BUGS  
/usr/share/doc/parted-2.1/BUGS  
/usr/share/doc/cvs-1.11.23/BUGS  
/usr/share/doc/patchutils-0.3.1/BUGS  
/usr/share/doc/procps-3.2.8/BUGS  
/usr/share/doc/gettext-0.17/BUGS  
/usr/share/doc/curl-7.19.7/BUGS  
/usr/share/doc/sed-4.2.1/BUGS  
/usr/share/doc/SDL-1.2.14/BUGS  
/usr/share/doc/cairo-1.8.8/BUGS  
/usr/share/doc/emacs-common-23.1/BUGS  
/usr/share/doc/tcsh-6.17/BUGS  
/usr/share/doc/unzip-6.0/BUGS  
/usr/share/doc/vsftpd-2.2.2/BUGS  
/usr/share/doc/dejavu-fonts-common-2.30/BUGS  
/usr/share/doc/nano-2.0.9/BUGS  
[rsimms@oslab ~]$
```

Use find to search for files by name, type, user, group, etc.

Finding Things

Task

Count all the files in the /home branch of the file tree that are owned by rsimms. Discard any permission errors.

```
find /???? -user ?????? 2> /dev/???? | ?? -1
```

Write your answer in the chat window

Finding Things

Task

Find all files in the */home/cis90/bin* that are regular files and belong to the staff group.

```
find /home/cis90/bin -group staff -type f
```

```
/home/cis90/bin/enlightenment
/home/cis90/bin/allscripts
/home/cis90/bin/list
/home/cis90/bin/submit.sp15.v1
/home/cis90/bin/tinsam90/schedule.pyc
/home/cis90/bin/tinsam90/schedule.py
/home/cis90/bin/tinsam90/forums.py
/home/cis90/bin/tinsam90/tips.py
/home/cis90/bin/tinsam90/grade.py
/home/cis90/bin/submitx
/home/cis90/bin/old/submit.fa14.v5
/home/cis90/bin/old/checkgrades.py.fa14
/home/cis90/bin/old/allscripts.sp14
/home/cis90/bin/old/check10.v2
/home/cis90/bin/old/submit.fa14.v1
/home/cis90/bin/old/check10.v1
/home/cis90/bin/old/submit.fa14.v4
/home/cis90/bin/old/checkgrades.py.sp14
/home/cis90/bin/old/submit.fa14.v2
/home/cis90/bin/old/submit.fa14.v3
/home/cis90/bin/old/submit.fa14.v6
/home/cis90/simben $
```

Use find to search for files by name, type, user, group, etc.

Finding Things

Task

Count all the directories in the `/home/cis90` branch of the file tree that belong to the cis90 group. Discard any permission errors.

```
???? /home/????? -type ? -group ????? ?? /dev/null | ?? -?
```

Write your answer in the chat window

Finding Things

Task

Find an account for simben90 in /etc/passwd

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

Finding Things

Task

Find your account in /etc/passwd

```
grep ??????90 /etc/passwd
```

Write your results in the chat window

Finding Things

Task

Find all files in the `/usr/src` branch of the file tree that contain "Torvalds"

`grep -r "Torvalds" /usr/src`

```

/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/powerpc/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/s390/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/parisc/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/alpha/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/boot/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/387.h: * Copyright (C) 1994 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/irq.h: * (C) 1992, 1993 Linus Torvalds, (C) 1997 Ingo Molnar
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/thread_info.h: * - Incorporating suggestions made by Linus Torvalds and Dave Miller
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/backops.h: * Copyright 1992, Linus Torvalds.
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/backtrace.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/hw_irq.h: * (C) 1992, 1993 Linus Torvalds, (C) 1997 Ingo Molnar
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/delay.h: * Copyright (C) 1993 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/include/asm/sync_bitops.h: * Copyright 1992, Linus Torvalds.
/usr/src/kernels/2.6.32-220.23.1.el6.i686/arch/x86/boot/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/loop.h: * Authors: Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/pagemap.h: * Copyright 1995 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/thread_info.h: * - Incorporating suggestions made by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/ext2_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/nfs2_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/dcache.h: * with heavy changes by Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/completion.h: * (C) Copyright 2001 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/id.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/ide.h: * Copyright (C) 1994-2002 Linus Torvalds & authors
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/delay.h: * Copyright (C) 1993 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/ext3_fs_sb.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/ext3_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/linux/ext2_fs_sb.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-220.23.1.el6.i686/include/asm-generic/tlb.h: * Based on code from mm/memory.c Copyright Linus Torvalds and others.
/usr/src/kernels/2.6.32-220.23.1.el6.i686/scripts/package/builddeb: Copyright: 1991 - 2009 Linus Torvalds and others.
/usr/src/kernels/2.6.32-220.23.1.el6.i686/scripts/get_maintainer.pl:push@penguin.chief: Linus Torvalds:torvalds@linux-foundation.org);
/usr/src/kernels/2.6.32-71.el6.i686/arch/powerpc/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/s390/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/alpha/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/alpha/boot/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/boot/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/irq.h: * (C) 1992, 1993 Linus Torvalds, (C) 1997 Ingo Molnar
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/thread_info.h: * - Incorporating suggestions made by Linus Torvalds and Dave Miller
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/backops.h: * Copyright 1992, Linus Torvalds.
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/backtrace.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/hw_irq.h: * (C) 1992, 1993 Linus Torvalds, (C) 1997 Ingo Molnar
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/delay.h: * Copyright (C) 1993 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/include/asm/sync_bitops.h: * Copyright 1992, Linus Torvalds.
/usr/src/kernels/2.6.32-71.el6.i686/arch/x86/boot/Makefile: # Copyright (C) 1994 by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/loop.h: * Authors: Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/pagemap.h: * Copyright 1995 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/thread_info.h: * - Incorporating suggestions made by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/ext2_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/nfs2_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/dcache.h: * with heavy changes by Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/completion.h: * (C) Copyright 2001 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/id.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/ide.h: * Copyright (C) 1994-2002 Linus Torvalds & authors
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/delay.h: * Copyright (C) 1993 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/ext3_fs_sb.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/ext3_fs.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/linux/ext2_fs_sb.h: * Copyright (C) 1991, 1992 Linus Torvalds
/usr/src/kernels/2.6.32-71.el6.i686/include/asm-generic/tlb.h: * Based on code from mm/memory.c Copyright Linus Torvalds and others.
/usr/src/kernels/2.6.32-71.el6.i686/scripts/package/builddeb: Copyright: 1991 - 2009 Linus Torvalds and others.
/usr/src/kernels/2.6.32-71.el6.i686/scripts/get_maintainer.pl:push@penguin.chief: Linus Torvalds:torvalds@linux-foundation.org);
/usr/src/kernels/2.6.32-71.el6.i686/scripts/checkstack.pl: # Inspired by Linus Torvalds
[rsim@ostlab ~]$

```

*Do a recursive grep to search the **contents** of files in an entire branch of the file tree.*

Finding Things

Task

Count the number of files in the `/usr/src` branch of the file tree that contain "Stallman"

```
grep -? "Stallman" /???/??? | wc -?
```

Write your answer in the chat window



Eggs, Treats and Tricks



trick or treat

A number of *trick* and *treat* files have been distributed within your home directory and sub-directories!

1. Can you find them? There should be an obvious one in your home directory. The rest are scattered in the various subdirectories you own.
2. Make a new directory named *bag* in your home directory and see how many *trick* or *treat* files you can move into it.

For the first several trick or treat files you find, cat the file, and write in the chat window what you got.

The write in the chat window when you get all six treats and then again when you get all 12 tricks and treats.

Review

Jim's Summary Pages

Jim has some really good summary information on Lessons 6-8 on his web site:

Lesson 6 - Managing Files

<https://web.archive.org/web/20100708145536/http://www.cabrillo.edu/~jgriffin/CIS90/files/lecture5.html>

Lesson 7 - File Permissions

<https://web.archive.org/web/20100708151130/http://www.cabrillo.edu/~jgriffin/CIS90/files/lecture6.html>

Lesson 8 - Input/Output Processing

<https://web.archive.org/web/20100708151725/http://www.cabrillo.edu/~jgriffin/CIS90/files/lecture7.html>



Make Teams

Make Teams



Everyone needs to be on ConferZoom today

Instructor:

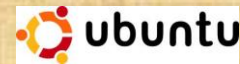
```
PS1="[u@h W] \t \$ "
```

```
/home/rsimms/scripts/teams (use teams alias)
```



Flashcard Practice

Flashcards



Flashcards

L6=20

L7=15

L8=16

Rules

- Chat window belongs to team that is up
- Team gets the point if anyone on the team writes a correct answer in the chat window in 15 seconds

Instructor timer:

```
i=15; while [ $i -gt 0 ]; do clear; banner $i; let i=i-1; sleep 1; done; clear; banner done  
(Use countdown alias)
```

Flash Cards

Click on Flashcards in left panel

Rich's Cabrillo College CIS Classes
Login Page

Home Resources Forums CIS Lab CTC

Login
Flashcards
Admin

CIS 90
CIS 192
Previous Classes

87 days till term ends!

Cabrillo College
Static IPs

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

Login
Flashcards
Admin

CIS 90
CIS 192
Previous Classes

87 days till term ends!

Cabrillo College
Static IPs

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 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)	

87 days till term ends!

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

Class Exercise Flashcards

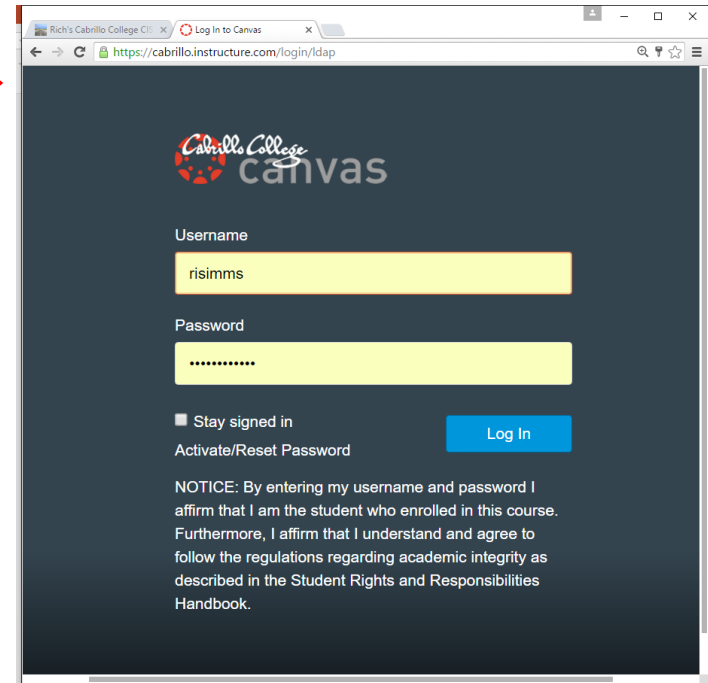
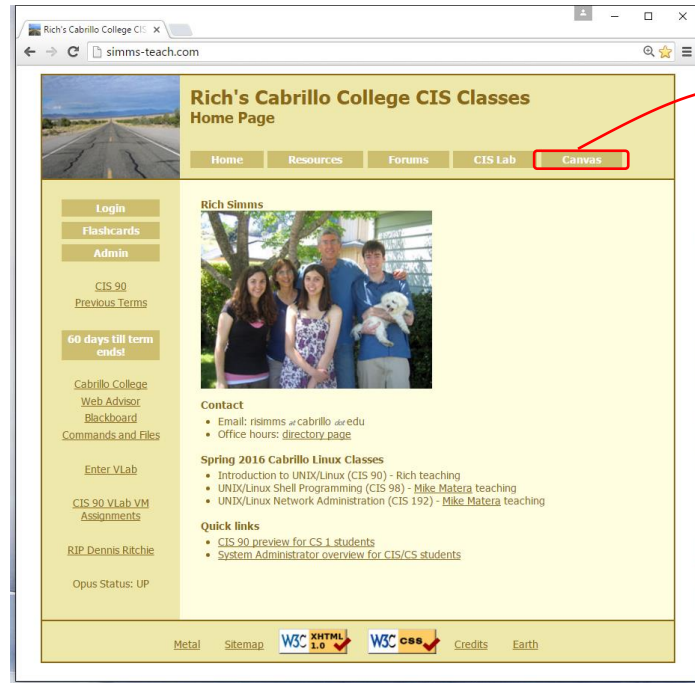
- Browse to simms-teach.com
- Register with a username and password of your choice
- Verify you can login and use the flash cards.

Assignment



Practice Test

Practice Test



Practice test available

- Available on Canvas
- Work alone or together in study groups
- Use the forum to compare answers and approaches to questions
- Test #2 will be graded by looking at both your answers to the questions and the work you did on the testing server.

Practice Test Instructions

HONOR CODE:

This is a practice test and you may work with others on it. You are encouraged to compare and discuss answers with your classmates using the forum, study groups or both. However on the real test you must work alone.

INSTRUCTIONS:

Test system: sun-hwa-p2.cis.cabrillo.edu (port 22)

This test should be completed using the sun-hwa-p2 system only. Because this system is on a private network, log into Opus-II first, then ssh into sun-hwa-p2. The practice test and the sun-hwa-p2 system will **not be available** after the real test starts.

Grading will be based on your answers AND that you correctly implemented the "DO THIS FIRST" portion of the question.

If you get stuck on a practice test question you can ask your classmates for help on the forum. If you get stuck on the real test and can't proceed you can ask the instructor for help and forfeit the point. For the real test the instructor will be available during class and available by email later in the evening from 8:00-10:00PM.

Please KEEP YOUR ANSWERS TO A SINGLE LINE ONLY !!

This is a practice test and unlike the real test you can take it as many times as you want. To prepare for the real test keep taking this practice test over and over again till you can answer each question in under 30 seconds.



P = 5 minutes before class ends (*noon or 4pm*)

T = when real test starts (*11am or 3pm*)

T-30 = 30 minutes before real test starts (*10:30am or 2:30pm*)

Reminder to instructor:

On Canvas

- Schedule Practice Test from **P** till **T-30**
- Publish Practice Test
- Moderate any accommodations
- Remove password on practice test
- Update test Q21 for number of accounts

On Practice Test system

- echo "/root/unlock-cis90; passwd -l cis90; rm /etc/nologin" | at **P**
- echo "/root/lock-cis90; cp /etc/nologin.bak /etc/nologin" | at **T-30**



Wrap up

Next Class

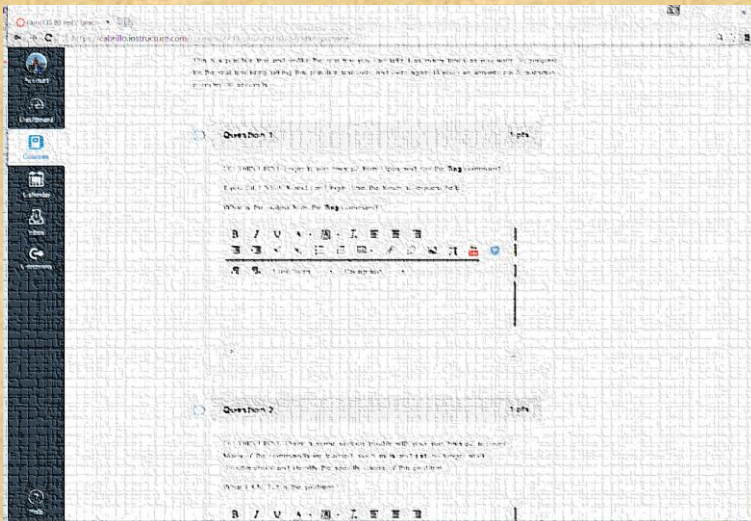
No Quiz

Test 2!

Cumulative Test (30 points) with focus on Lessons 6-9:

- Recommended preparation:
 - Work the practice test!
 - Restore your directory with `./restore` and work the practice test again!
 - Repeat step above till you can answer all questions in 30 seconds
 - Make a personal reference "crib sheet" document
 - Collaborate with others on the forum to compare answers
 - Review Lessons 6-9 slides and Labs 5-7
 - Try doing some or all of Lab X2 (pathnames)
 - Practice with flash cards
 - Scan previous Lessons so you know where to find things if needed

If time permits



Work the practice test till the end of class.

- Collaborate!
- Ask questions!
- Arrange study groups!

Backup