Getting Started

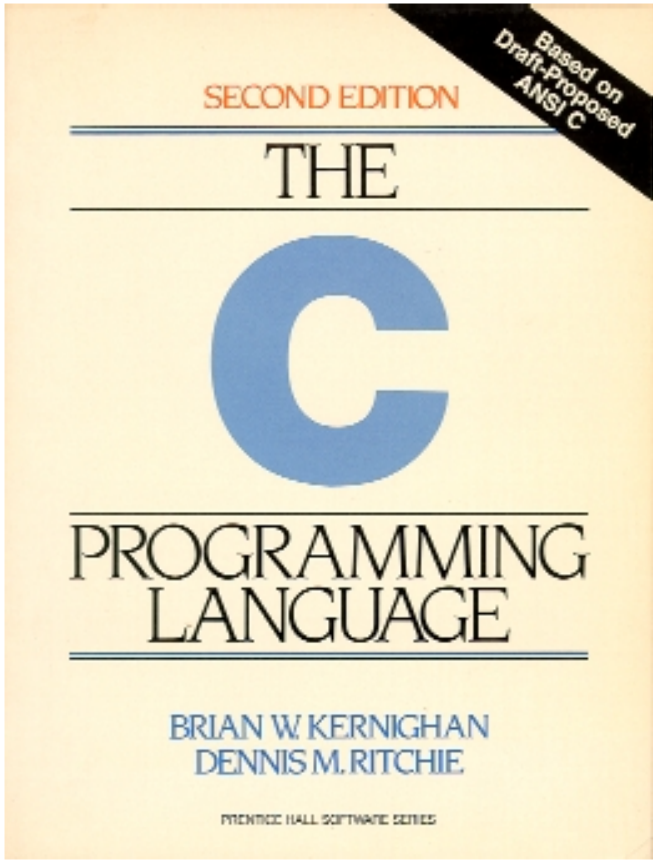
Andrew Christlieb (Instructor)

D319 Wells Hall

christli@msu.edu

office hours

Thursday 3-5


SECOND EDITION
Based on Draft-Proposed ANSI C
THE C PROGRAMMING LANGUAGE
BRIAN W. KERNIGHAN
DENNIS M. RITCHIE
PRENTICE HALL SOFTWARE SERIES

Classroom: 3353 Engineering Building

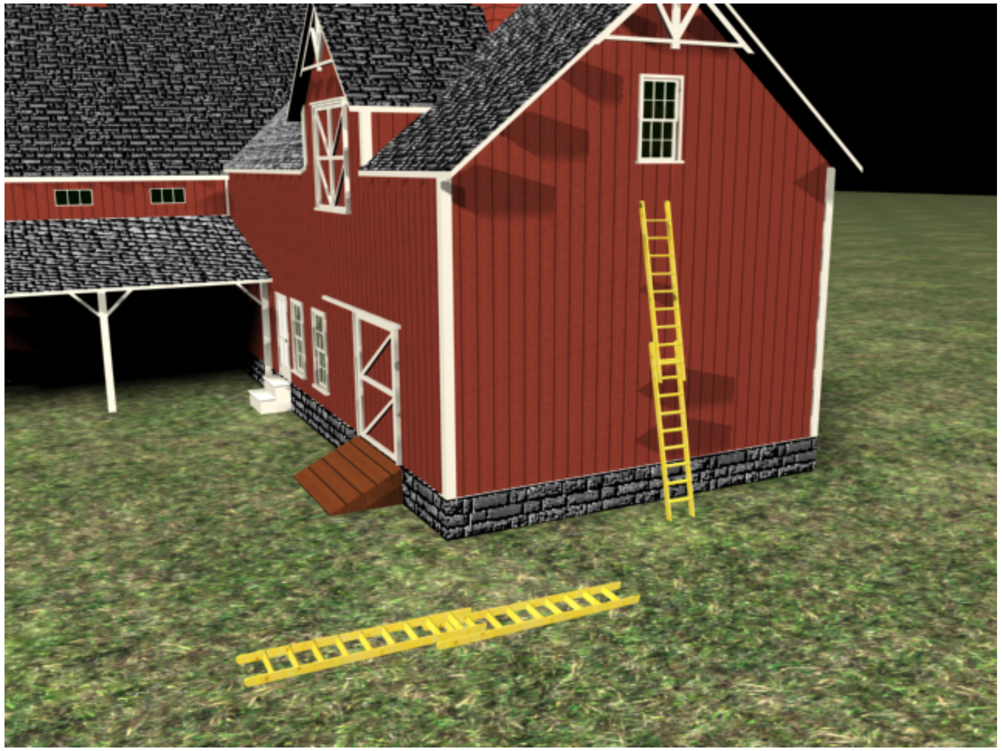
Structure of the Course

Get a quick intro to C programming

Learn enough Unix to use it

Get some experience with C-oriented tools.

Get some experience writing code

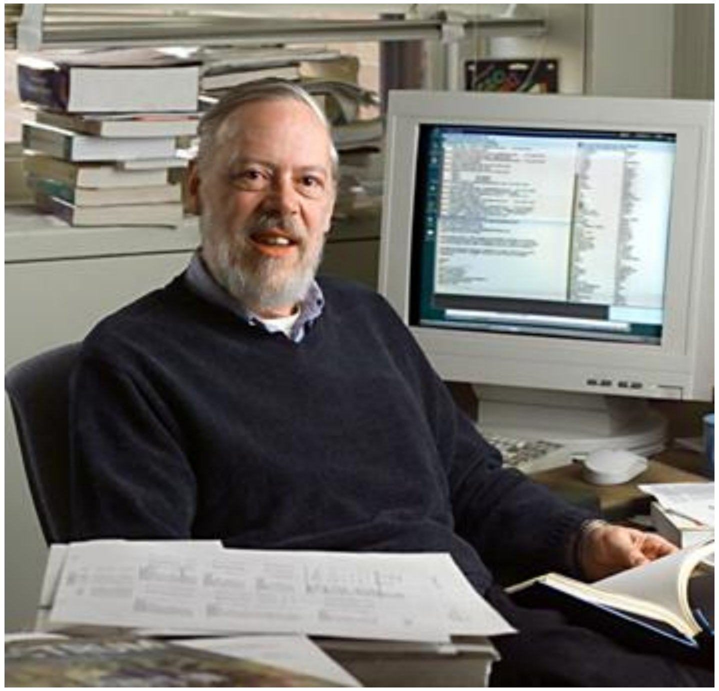


What is C?

C is a rather old programming language (1972, Richie, Bell Labs)

Originally designed as a systems software platform (OS and the like)

Procedural, block oriented language (no object-oriented programming)


Dennis Ritchie
Inventor of the
C Programming Language

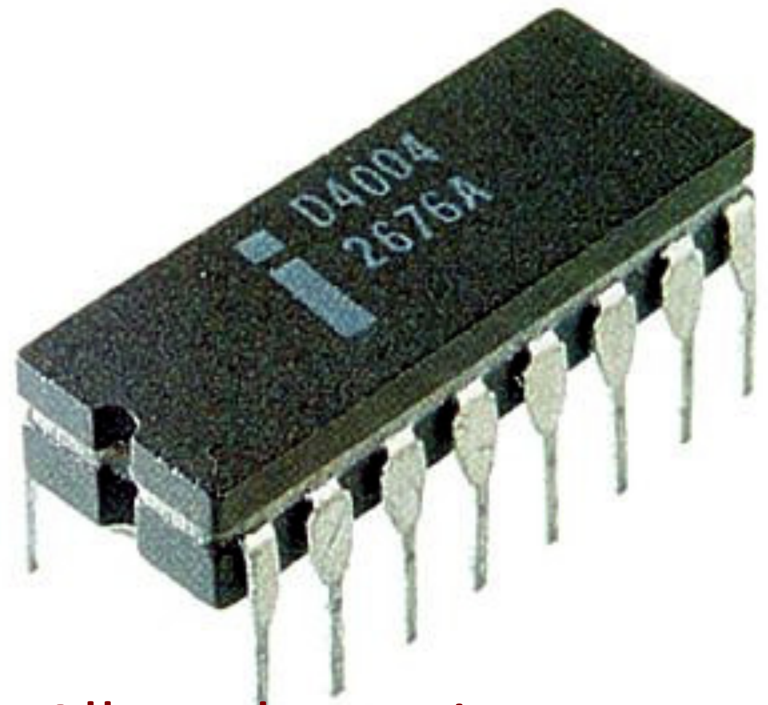
Why learn C?

- Small, extensible language. Progenitor of many languages.
- Many applications and much support due to its age and general use
- Many tools written to support C development.
- Close to the hardware, programmer manages memory
- Common embedded systems language
- Can be fast, efficient (most cited reason)

Disadvantages of C?

- Flexible language allows programmers to commit many sins without warning
- Hard to debug, fix code
- Speed depends on programmer as much as the language
- Managing memory can be dangerous, difficult, painful
- Lacks modern features (OOP, exceptions, etc.)

Why do you care? Programming Microprocessor



All modern micro possessors are designed to be programmed via C. **Industry Standard!**

Applications:

- Robotics
- Controller Area Networks
    - Automotive
    - Caterpillar
    - Aircraft
- Video Gaming (GPUs)
- High End Computing For Comp EM (GPGPUs)
- Adaptive Controls
- Remote Sensing (Bridge Failure?)
- Digitally Controlled Adaptive Filters
- ALL OF ECE IS IMPACTED!

Course Structure

Course will be Lecture/Step

Sep-13						
Su	Mo	Tu	We	Th	Fr	Sa
1	**2**	3	4	**5**	**6**	7
8	**9**	10	11	**12**	**13**	14
15	**16**	17	18	**19**	**20**	21
22	**23**	24	25	**26**	**27**	28
29	**30**					

Oct-13						
Su	Mo	Tu	We	Th	Fr	Sa
		1	2	**3**	**4**	5
6	**7**	8	9	**10**	**11**	12
13	**14**	15	16	**17**	**18**	19
20	**21**	22	23	**24**	**25**	26
27	**28**	29	30	**31**		

Nov-13						
Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	**4**	5	6	**7**	**8**	9
10	**11**	12	13	**14**	**15**	16
17	**18**	19	20	**21**	**22**	23
24	**25**	26	27	**28**	**29**	30
	Monday			**Wells D319 3-5pm**	**Friday**	

Dec-13						
Su	Mo	Tu	We	Th	Fr	Sa
1	**2**	3	4	**5**	**6**	7
8	**9**	**10**	**11**	**12**	**13**	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				
	Monday			**Wells D319 3-5pm**	**Friday**	

<u>This is a 1 credit course, meets only 14 times!</u>

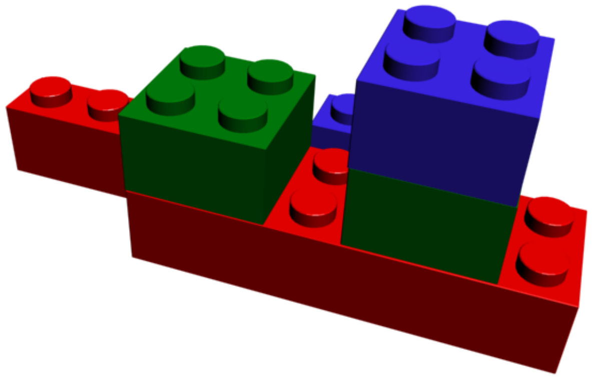
<u>Lecture/Step approach</u>

Spend some time on a lecture topic.

Spend some time doing "hands-on" work.

lecture/step will have some exercise you will have to turn in at the end

Grading



3 projects, each for 15% of the grade, total 45%

13-15 Step assignments, exercises, total 55%

That's it. No exams, no quizzes

Attendance is important

We drop the lowest step assignment grade!

Should go without saying that in a course like this attendance is important.

Step exercises

- These can be collaborative, done with discussion and help from anyone during lab. Talk to people, ask questions.

- Lab time is a time to figure stuff out

Projects are individual

- No collaboration on projects
- They will be checked by cheat check
- Just don't do it. Do your own work

- **No makeups**! There are only 3 and I'll give plenty of notice. Get it in on time.

Get Started, booting

Rebooting to linux

- Machines are split to run either linux (a unix variant) or Windows
- You can reboot to run “diskless” and bring any machine up in linux.
- Let’s do that now!

Booting

If your machine says "press control-alt-delete to log in", do so, but use the red button in the lower right corner to shut the machine down. Then restart, selecting "diskless" when prompted.

Before logging in, use the Session menu to select GNOME as your session.

Once logged in, select Applications/Internet/Iceweasel Web Browser.

Go to http://www.cse.msu.edu/~cse251

Initial password is your PID (starts with A)

0

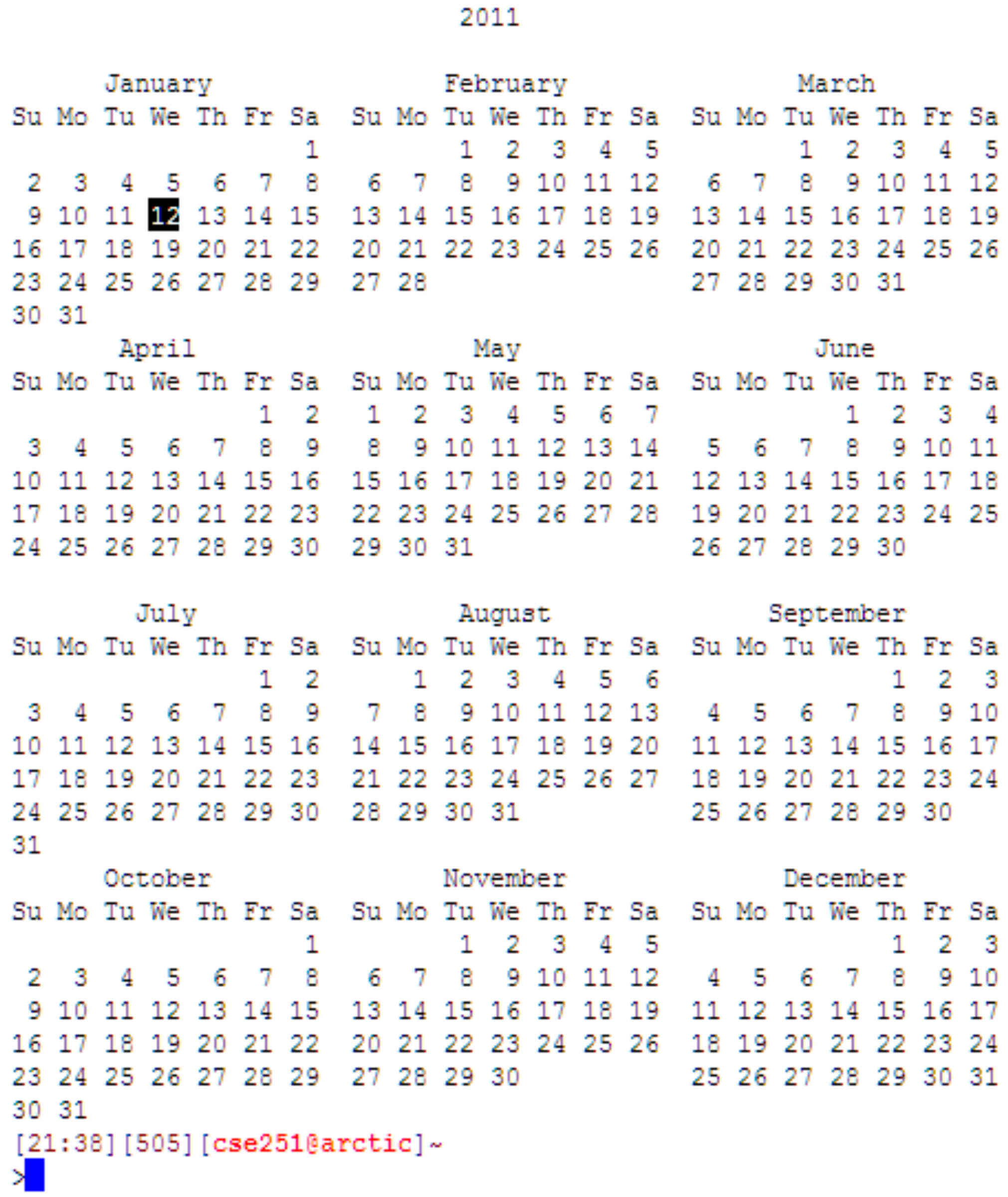
Bringing up…

Bring up a terminal Window

Applications/ Accessories/Terminal

Type:

cal 2011



```
                               2011

       January               February                 March
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
                   1          1  2  3  4  5          1  2  3  4  5
 2  3  4  5  6  7  8    6  7  8  9 10 11 12    6  7  8  9 10 11 12
 9 10 11 12 13 14 15   13 14 15 16 17 18 19   13 14 15 16 17 18 19
16 17 18 19 20 21 22   20 21 22 23 24 25 26   20 21 22 23 24 25 26
23 24 25 26 27 28 29   27 28                  27 28 29 30 31
30 31
        April                   May                   June
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
                1  2    1  2  3  4  5  6  7             1  2  3  4
 3  4  5  6  7  8  9    8  9 10 11 12 13 14    5  6  7  8  9 10 11
10 11 12 13 14 15 16   15 16 17 18 19 20 21   12 13 14 15 16 17 18
17 18 19 20 21 22 23   22 23 24 25 26 27 28   19 20 21 22 23 24 25
24 25 26 27 28 29 30   29 30 31               26 27 28 29 30

        July                  August                September
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
                1  2       1  2  3  4  5  6                1  2  3
 3  4  5  6  7  8  9    7  8  9 10 11 12 13    4  5  6  7  8  9 10
10 11 12 13 14 15 16   14 15 16 17 18 19 20   11 12 13 14 15 16 17
17 18 19 20 21 22 23   21 22 23 24 25 26 27   18 19 20 21 22 23 24
24 25 26 27 28 29 30   28 29 30 31            25 26 27 28 29 30
31
       October               November               December
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
                   1          1  2  3  4  5                1  2  3
 2  3  4  5  6  7  8    6  7  8  9 10 11 12    4  5  6  7  8  9 10
 9 10 11 12 13 14 15   13 14 15 16 17 18 19   11 12 13 14 15 16 17
16 17 18 19 20 21 22   20 21 22 23 24 25 26   18 19 20 21 22 23 24
23 24 25 26 27 28 29   27 28 29 30            25 26 27 28 29 30 31
30 31
[21:38][505][cse251@arctic]~
>
```

Directories

What is a directory/folder?

<u>Be it Windows, Linux or OS X, all OS's maintain a directory structure.</u>

A directory is a container of files or other directories

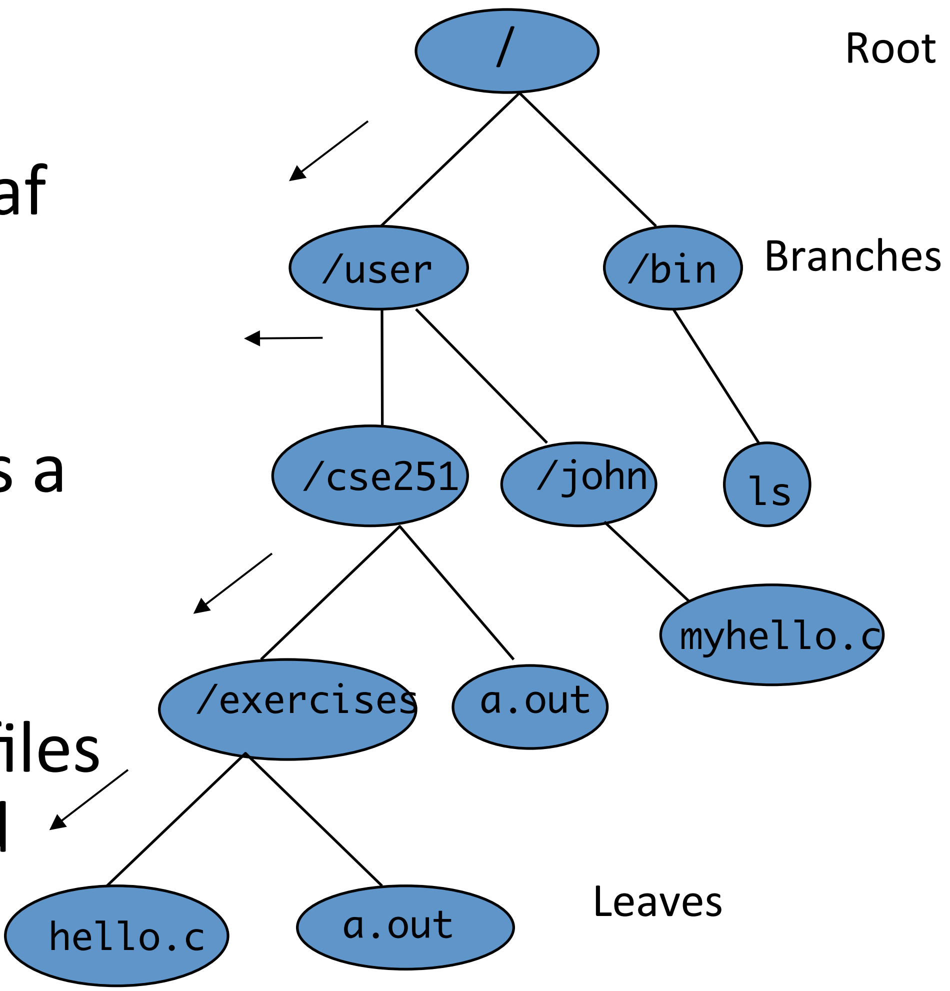
These directories are arranged in a hierarchy or tree

Directory Structure

Has a root node, with branch nodes, ends in leaf nodes

The directory structure is a tree

Each directory can hold files and 'point to' parent and children directories

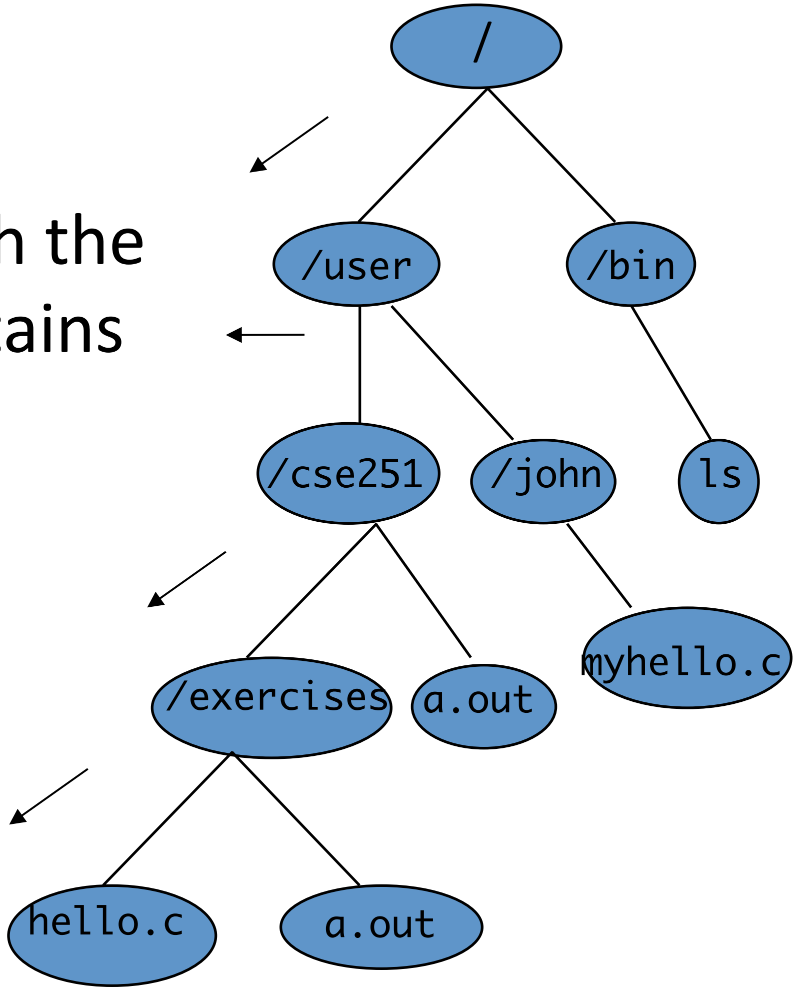


File Path

A path to a file is a path through the hierarchy to the node that contains a file or directory

/user/cse251/exercises/hello.c

Path is from root node /, to user directory, to cse251 directory, to exercises directory, where the file hello.c resides.



The current directory

As you type in the command shell, you are presently "in" one current directory

Many commands are available to navigate by changing your current directory

Unix command caution(s)

Names are short. This is to avoid typing. Get used to it

Most commands take “switches”, parameters which modify the behavior of the command. Usually preceded by a ‘-’

All commands, all switches, all paths, all filenames are <u>case sensitive</u>

Some Unix Commands

pwd – Displays the current working directory

ls – Lists the contents of the current directory

cd – Changes the current directory

mkdir – Creates a new directory

Path String

a valid path string is a series of directories (separated by ‘/’) which indicates a valid path in the directory structure

‘/user/cse251/exercises/hello.c’ is a valid path string but ‘/cse251/hello.c’ is not

Three special directory names

- ‘.’ is shortcut for current directory you are in
  - ‘./a.out’ is the same as ‘/user/cse251/exercises/a.out’ if you’re currently in /user/cse251/exercises directory
- ‘..’ is shortcut for the name of the parent directory of the current directory you are in
  - ‘../a.out’ is the same as ‘/user/cse251/a.out’ if you’re currently in /user/cse251/exercises directory
- ‘~’ is a shortcut for your home directory
  - ‘~/myhello.c’ is the same as ‘/user/john/myhello.c’ if your login name is john

mv (changes a file)

mv is the move command. Rename a file in place or move to a new directory

- mv file.c newFile.c
  - rename file.c to newFile.c in current directory
- mv file.c /user/ptan/
  - move file.c in the current directory to /user/ptan. The target directory must already exist
- mv ~/file.c ./newFile.c
  - move file.c in home directory to the current directory with the name newFile.c

cp, copy a file

- cp file.c newFile.c
  - Create a new file called newFile.c in current directory from the file file.c, which is also in current directory
- cp ../file.c ~/programs
  - cp file.c in parent directory to the sub-directory programs under the home directory

rm, remove a file

Cannot be undone, be careful

- rm /user/cse251/exercises/a.out
  - remove the file called a.out
- rm ./file.c
  - remove file.c in the current directory
- rm –i ./file.c
  - interactive, are you sure?

man

man pages exist on Unix, providing documentation (a lot of documentation) on each command

- man ls
  - man page on the ls command
- man -k graphics or apropos graphics
  - every man page that has “graphics” as a word
- man -S 2 mount
  - man page for section 2 of mount

Better man pages

- Almost every distribution of Linux provides a better way to view man pages
- On Debian/Ubuntu, the program yelp (the help program) will show man pages
- yelp, then man ls
  - show man page on ls in a nice way
- yelp runs using the “lifesaver” icon. Go to Advanced options, man pages

What’s the shell

- the shell is the program that interacts with you through the terminal window
- there are many, and you can change it easily
  - csh
  - ksh
  - tcsh
  - bash
- by default, you are using tcsh

Prompt

Default prompt looks like:

- <47 nelson:/usr/include >
  - 47, which command in the history this is
  - nelson, the name of the machine you are on
  - /usr/include, current directory

can be configured differently

- [19:19][31][cse251@nelson]~
- >
  - two lines: time, history, who@machine, path

Job control

Only one job runs in foreground (sending text to console), but many can run in background

Foreground job controls the console

- emacs myFile.c &
  - & mean run emacs in background. Console remains responsive to commands
- jobs
  - lists all jobs running in background. Numbers can be used to control job

ps: List of processes

- Typically, each job that is executing is called a process
  - Sometimes, a job can produce (fork) multiple processes

- Use "ps" command to list all your current processes (including those that were suspended or running in background)
  - It also shows the background processes (including the shell program you're using)

Command completion

Type a partial command, followed by tab. it will complete as much as it can

Compiling

compile to make an executable

- In C, you cannot directly execute the source code. You go through a process called compilation
- Compilation makes sure you “followed the rules of C” (even if you did something stupid), and makes an executable
- Compilation errors are rule “mistakes”. A compilation error means no executable
- The executable file is the thing you run

gcc: gnu compiler

- There are others, but this is a good one and it is free (and on all Linux distributions, can be added to the MacOS and runs under Cygwin on windows)

- Has just around a zillion switches. Can be pretty impressive if you read the man page.

Example gcc usage

- gcc file.c
  - if it compiles, makes an executable called a.out in the same directory as file.c
  - can run a.out by typing ./a.out (meaning the a.out in the current directory)
- gcc -o myExec file.c
  - make the executable called myExec

Compilation errors

- You didn't follow the rules
- You have to fix the source code before the compiler can make an executable
- Errors can be cryptic, but at the very least they list the line number where it went wrong
- Doesn't prevent run-time errors

Editors

Many Editors

Examples:

- gedit
- vi
- emacs
- pico
- kate

I'm going to suggest using gedit in this course.