CptS 223 - Advanced Data Structures in C++

Fall 2017 - Introduction and Day 1
Instructor: Aaron S. Crandall, PhD

Today’s Plan -
Introduce the Course and other Things

- Greetings & Who am I?
- Attendance (yes, it will take a while)
- Syllabus Review
- The challenges of this course
- Quick ungraded knowledge quiz via Google Drive
- Course contents, textbook, & topics
- Motivation for this course
- Resources available for help during term
- My lecture style
- More questions

Who is this Instructor? - Aaron Crandall

- Academic credentials: Phd (WSU, 2011), MSCS (OHSU, 2007), BSEE (UP, 2001)
- Research credentials: Publications on smart homes, UX, gerontechnology, AI
- Teaching credentials: Linux IT Basics, Gerontechnology, Security Basics
- Industry credentials: Startup in gerontech, IT (linux, win, solaris, wifi) 5+ years
    - Software engineering @ couple of companies, telecommunications
- WSU Clubs: Cougs in Space, Ham Radio, Computer Security Group, Hurling @ WSU
- Other stuff: Aerospace Club (rocket building), drum major for pipe band
    - Arduino projects, LUG events (WSUCon), ACM events (Hackathon)
- Also teach CptS 421/423 - Senior Design / Capstone
- Been at WSU since 2006 (FWIW)
- Linux user since 1997, though no guru by guru standards

Announcements

- Course is managed via Blackboard
- Assignments will be turned in by checking them into your Git repository and submitted in the checkin hash to Blackboard so the TA can find it
- Our TA's are not fully assigned yet, but we're working on it
- Linux User's Group is here to help you - tutoring and office hours
- Winter WonderLAN is coming up this fall - LAN Party by the LUG
- Encouraging people to look at Frank Fellows program
  - https://goo.gl/VKrVQC

We’re going to take attendance!

- Yes, I’m going to pronounce many things incorrectly, but I’ll do my best
- Yes, this is going to take a while
- It’s so I can get a sense of names to faces
- When we do this for credit later on, we’ll use a faster tool
    - Actually, I use Google Forms for attendance.
    - Make sure to bring a web-capable device to class: laptop, smartphone, etc.
    - Can borrow one from a neighbor if need be
- How to teach Google Home to pronounce your name: https://www.cnet.com/how-to/google-home-teach-google-how-to-pronounce-your-name/

Syllabus review

- Current syllabus
- Need to review the course, grading, goals, and general topics

  Syllabus

- 223 open labs - TBD as final adjustments are made by the department

The Challenges of This Course

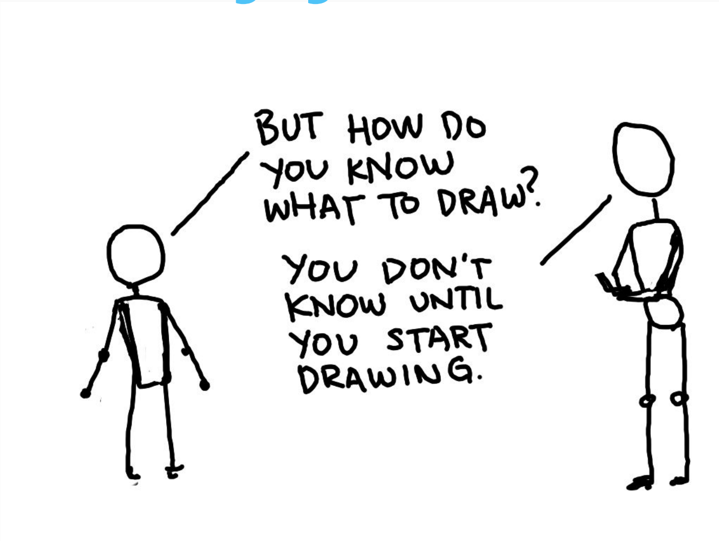
- This course is challenging on a few fronts:
  - Core content: Data Structures
  - Theory introduction: Algorithmic Complexity & Analysis
  - Coding environment: Linux with gcc/make
  - Course structure: No official lab section for built-in help
- That said, we can do this!
  - I’m here to help (since that’s my job and all!)
  - There’s still lots of resources available
  - This stuff can actually be quite fun
    - But fun normally begins after skill, not before!

Do you feel inspired yet? I could find a picture of a majestic eagle if that would help.

Quick (Ungraded!) Knowledge Quiz via Google Docs

URL for quiz: http://bit.ly/2wxjtjE

- This quiz is designed to get me a handle on where people stand
- CptS 122 covers the basics of data structures
- I'd like to make sure we gracefully transition into more advanced material
- Please follow this URL on your digital device of choice (laptop, tablet, phone)


BUT HOW DO YOU KNOW WHAT TO DRAW?
YOU DON'T KNOW UNTIL YOU START DRAWING.

Course Contents (A general taste)

- Algorithm Analysis:
  - How long will the algorithms A or B take given input size n?
- Data Structures:

Lists, Stacks, Queues	Trees	Hashing
Graph Algorithms	Sorting (sorting, sorting!)	More advanced stuff

- Linux use (yes, you'll get to learn a new OS environment)
  - It's a lot less scary than you think. It also runs your universe, so it's good to be friends
- Git for version control - so very useful
- C++ STL makes your life easier! Also... gtest (if I can make it work for us), lots of techie things (ddd, valgrind), plus Life, the Universe and Everything.

Motivation for This Course:
Or, why require this course for CompSci?

- Here's at least 3 biggies that come to mind:
    a. After basic programming comes true data handling
    b. Choices you make in computation algorithms and data storage have huge impact

       This skill is what sets you apart from someone who can just code!

    c. Linux gcc/make + git + testing are keystones to modern computing behind the scenes

Resources for Help During Term

There is no substitute for practice!

Resources for Help During Term

That said, here's some other resources:

- Weekly EECS open labs for homework help from TA's (time & place TBD)
- Saturdays with the Linux User's Group - programming and Linux training
- I have office hours! - See Syllabus
  - EME 506
- The TAs will have office hours! (TBD)
- The textbook actually *does* cover the material if you read it
  - Fallbacks include Google and Wikipedia
- Study groups can help. They've saved my bacon more than once.

My Lecture Style - Yes, it gets its own slide

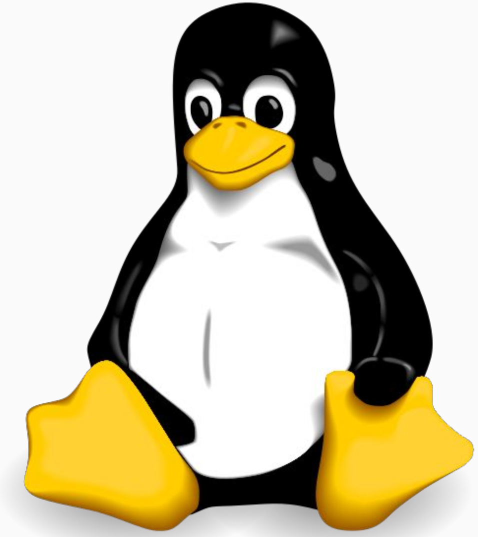
- I have an interruptible style
  - I don’t have my lectures timed out to the second
- I’m open to questions at any time
  - I’ll let you know if I don’t have time for them
  - Start out asking and we’ll see how it goes
- I’m going to assume you’ve read any materials I’ve assigned you
  - Questions clarifying material are really cool
  - Questions covered in readings… perhaps not so cool

More Questions?

- You’re going to do great if you:
  - Attend class
  - Do the readings
  - Do the homeworks
  - Practice coding
  - Get help if you’re stuck
  - Trust me when I say the command line won’t kill you
  - Pay attention!
  - Be curious. Often the best way to figure something out is to write a code snippet to test the thing that’s got you hung up. It’s entirely okay to write throwaway tiny programs. The computer won’t mind.

Next Class: Intro to Linux/Initial Demos

- I’ll follow up with any unanswered questions
- We’ll do some basic gcc/make + git use materials
- The first assignment is going up on Blackboard
  - Get your EECS ISG login working: support@eecs.wsu.edu
  - Login to the EECS SSH server
  - Copy the demo code there, compile it, screenshot and upload
  - You’ll need to learn:
    - SSH, SCP, make/g++, screenshots, tar, file handling
  - Due Monday. If you’re completely stuck, get help early!



This week’s reading: Preface & Chapter 1

- Please start on Weiss chapter 1
    - The goal is to be done by next Monday
    - Yes, there’s some math
    - Yes, there’s some programming (mostly getting you up on the STL in C++)
    - Yes, it’s time to get back to school. Break’s over (boo!)
- Assignment for Monday: Get logged into a Linux system + the Git Server
        - EECS has some available via SSH or NXClient
            - Your EECS account is in good order, right?
        - Alternatively, set up VirtualBox on your own computer
        - Doubly alternatively, set up a dedicated Linux box (dual boot or not)
        - LUG has help on Saturday, but you’ll need your EECS account before then!

Final Questions?

- I’ll do some gcc/g++ building if you (or I) leave me time