What This Class Is About

1. How a program executes on the hardware
2. The systems costs of program execution
3. An introduction to operating systems
4. Foundations of parallel programming
Instructor: Martin Gagné

- [http://www.cs.swarthmore.edu/~mgagne1/](http://www.cs.swarthmore.edu/~mgagne1/)
- Please call me Martin (or Professor Gagné)
- Research: Cryptography and Computer Security
- Hobbies: Sci-fi and Fantasy books, piano, PC games
Office Hours

• Monday 10:30 AM – 12:00 PM
• Tuesday - Thursday 2:00 – 3:30 PM
• Limited availability on Fridays
• By appointment

• 251 Science Center
Ninjas!

- Sessions Sundays 7-11 PM in 256

Charlotte

Douglass

Hunter

Lu Min
Tonight (& tomorrow)

• Using Unix / Linux help session
  – 7:00 PM – 8:00 PM
  – Open to everyone
  – If this is your first CS course here, you should go

  – Location: SCI 256 (I’m not 100% sure)
Resources

• Piazza Q&A Forum
  – https://piazza.com/swarthmore/spring2016/cs31/home

• Slides on course website

• Lab sections:
  – Clothier 016
  – Wednesdays 9:50-11:20, 1:15-2:45, 3:00-4:30
Email Policy

• Please use Piazza rather than email
  – Your classmates benefit from your questions
  – Your classmates can answer your questions
  – I will check the forum frequently

• I will attempt to respond to within 24 hours

• If you do email me, please use mgagne1@cs...
How does this class work?

• This class is designed a bit differently from what you might normally be used to
  – Class will be centered around discussion
  – Requires your participation

• Ever considered why we have lectures?
Traditional Lectures:

- Roughly one millennium old
Traditional Lectures:

- Little opportunity for expert feedback
- Might as well skip class and watch video lectures!
  - (I am not actually suggesting this. Please attend your classes!)
Interactive Classes with Peer Instruction

- You do the “easy” part before class.

Textbook, videos, website → In class quiz → Instruction → Exam
First Exposure → Gauge understanding → Fill in gaps, Explore details, Add context, Provide feedback → Show Knowledge Mastery

- Class is reserved for interactive, customized experiences
- Research on how people learn:
  - Everyone constructs their own understanding
  - To learn, YOU must actively work with a problem and construct your own understanding of it
Clickers!

- Lets you vote on questions in real time.
- Like pub trivia, except the subject is always systems.
- Please turn them off at end of class...
Peer Instruction

• Short quiz at the beginning of class
• During class: pose carefully designed questions
  – Solo vote: Think for yourself and select answer
  – Discuss: Analyze problem in teams of 3
    • Practice analyzing, talking about challenging concepts
    • Reach consensus
    • If you have questions, raise your hand and I’ll come over
  – Group vote: Everyone in group votes
    • You must all vote the same to get your point
  – Class wide discussion:
    • Led by YOU (students) – tell us what you talked about in discussion that everyone should know!
Why Peer Instruction?

• You get a chance to think.
• I get feedback as to what you understand.
• It’s more engaging!
• Research shows it promotes more learning than traditional lecture.
Example Question

• Individual vote

• Group discussion / group vote
  – Room should be LOUD

• Class discussion
The best TV series is:

A: Arrested Development
B: Breaking Bad
C: Firefly
D: The Wire

E: Some other series (be prepared to discuss what and why!)
Clicker Registration

- [https://clickers.cs.swarthmore.edu](https://clickers.cs.swarthmore.edu)

- Can register for course or one-day loaner

- This is how I take attendance this week (important for your registration)
Reading Quizzes

• Readings from online sources

• Target low difficulty: did you read?

• Goal: incentivize / reward preparation
  – Can be an easy 5%!

• You may bring handwritten notes.
Grading

• 5% Reading Quizzes
• 5% Class participation
• 5% Written Assignments
• 20% Midterm Exam
• 30% Final Exam
• 35% Lab Assignments
Grading

• 5% Reading Quizzes
• 5% Class participation
• 5% Written Assignments
• 20% Midterm Exam
• 30% Final Exam
• 35% Lab Assignments

• I will drop your three lowest quizzes/no-shows.
Supplemental Textbook

Policies

• Collaboration
  – You may discuss approaches, not solutions
  – You must submit your own work
  – Exams may include questions on programming

• Cheating
  – Zero tolerance for cheating, don’t do it!

• Lab / Assignment Lateness
  – 3 free late days for the semester
Tentative Schedule

• Midterm – March 2, in class

• Final - TBD

• Labs
  – Out on Wednesdays (lab section)
  – Due on Tuesdays
A Bit More on Labs

Labs
- Out on Wednesdays (lab section)
- Due on Tuesdays

Office Hours
- Monday 10:30 AM – 12:00 PM
- Tuesday - Thursday 2:00 – 3:30 PM
- Limited availability on Fridays
- By appointment

- 251 Science Center

Ninjas!
- Sessions Sundays 7-11 PM in 240

Charlotte
Douglass
Hunter
Lu Min
Administrative Questions?

- All of this info (should be) on class website
- Feel free to ask on Piazza discussion board
What is a computer system?

• Hardware and/or software that...
  – allows the user to interact with programs
  – allows programs to run and use machine’s *resources*
  – makes computer easier to use

• Improves the computer’s capabilities
  – performance
  – reliability
  – security
  – usability
Turn undesirable into desirable

• Hide undesirable inconveniences: reality
  – Complexity of hardware
  – Single/limited number of processors
  – Limited memory

• Project desirable conveniences: illusions
  – Simple, easy-to-use resources
  – Multiple/unlimited number of processors
  – Large/unlimited amount of memory
Three big ideas

• Abstraction
  – What is the desired illusion?
  – How do we interact with it?

• Mechanism
  – How do we create the desired illusion?
  – How does it work?

• Policy
  – How do we make it work well, to meet a goal?
Why should you care?

• To know how your computer works
  – What may be wrong with your programs
  – How to enhance your computer, applications

• Systems programmers get respect
  – In high demand, get paid well

• Real-world impact
Pacman

• Pacman freaks out if you complete level 255

• Why?
Mars Pathfinder (1997)

- Frequently locked up and stopped responding  
  - (automatic reboot)

- “Priority inversion” in parallel software
Pokémon Yellow

- Cleverly “hacked”, game completed in 1:36
- “Buffer overflow” exploit
Why should you care?

• Systems is a classic area of CS
  – Roots are in the early 60’s
  – Yet, still very relevant today

• Intellectually very challenging
  – Models
  – Programming
  – Practice
This Course

• How your programs *really* execute

• 1\textsuperscript{st} half: focus on hardware execution
• 2\textsuperscript{nd} half: focus on operating system
The scheduled office hour times are...

Monday 10:30 AM – 12:00, Tu - Th 2:00 – 3:30 PM

A. Great, I don’t have a scheduling conflict with either one.

B. Probably ok, I don’t have a scheduling conflict with at least one of them.

C. Not good, I have a scheduling conflict with all of them!

D. I’m not sure, I don’t know my scheduled yet.
The majority of my prior programming experience is in...

A. Python (CS 21)

B. Python (Some other source)

C. C or C++ (CS 35 or other)

D. Java

E. Some other language
Your TODO list

• Reading posted on course web page

• Sign up for Piazza!

• Please let me know (emails ok) about:
  – Your preferred name, if different than roster name
  – Your preferred gender pronoun
  – Disability accommodations

• Register your clicker, if you didn’t already…
Lab Switching

- Some of you contacted me to switch lab section
- Due to limited size, we can only organize swaps
- 1:15-2:45 is the most popular section, so it’s hard to swap into
Let’s get started!
How does a computer run your program?

1. Compiler translates c code into an executable.
2. Shell forks a new process.
3. Operating systems allocates resources to the process.
4. CPU loads instructions and data from memory.
5. Instructions specify calculations to perform on the data.
6. Current passing through circuits carries out calculations.
7. Circuits are composed of gates, which are built from wires and transistors.
This class builds from the bottom up

Order of systems topics:
• Binary representation of data
• Building simple circuits from gates
• Building a CPU from simple circuits
• Assembly language
• Memory
• Operating systems
• Processes
• Parallel Programming

We’ll also learn lots of C programming along the way.
Binary numbers

• How computers represent all data.

• Strings are represented as a sequence of characters, and each character is represented by a number.

• The screen is a collection of pixels, and each pixel’s color is represented by several numbers.

• All numbers are in binary: they’re made up of ones and zeros.
Let’s start with what we know...

- Decimal number system (Base 10)
- Sequence of digits in range [0, 9]

64024

Digit #4  Digit #0
Positional Notation

• The meaning of a digit depends on its position in a number.

A number, written as the sequence of digits \(d_n d_{n-1} \ldots d_2 d_1 d_0\) in base \(b\) represents the value

\[d_n \times b^n + d_{n-1} \times b^{n-1} + \ldots + d_2 \times b^2 + d_1 \times b^1 + d_0 \times b^0\]
Decimal: Base 10

• Used by humans

A number, written as the sequence of digits \(d_n d_{n-1} \ldots d_2 d_1 d_0\) where \(d\) is in \{0,1,2,3,4,5,6,7,8,9\} represents the value:

\[d_n \times 10^n + d_{n-1} \times 10^{n-1} + \ldots + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0\]

64024 =
6 \times 10^4 + 4 \times 10^3 + 0 \times 10^2 + 2 \times 10^1 + 4 \times 10^0

60000+ 4000 + 0 + 20 + 4
Binary: Base 2

• Used by computers

A number, written as the sequence of digits 
\[d_n d_{n-1} \ldots d_2 d_1 d_0\] where \(d\) is in \(\{0,1\}\), represents the value

\[d_n \times 2^n + d_{n-1} \times 2^{n-1} + \ldots + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0\]
Converting Binary $\rightarrow$ Decimal

• Two methods:
  • powers of two and addition
  • multiplication by two plus position bit
Method 1: powers of two and addition

E.g. start with binary number 100101

\[ 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \]

\[ = 1 \times 32 + 1 \times 4 + 1 \times 1 \]

\[ = 37 \]
Method 2: multiplication by two plus position bit

E.g. start with binary number 100101
What is the value of 110101 in decimal?

A. 26
B. 53
C. 61
D. 106
E. 128
Your TODO list

• Reading posted on course web page

• Sign up for Piazza!

• Please let me know (emails ok) about:
  – Your preferred name, if different than roster name
  – Your preferred gender pronoun
  – Disability accommodations

• Register your clicker, if you didn’t already…