CPSC 313 — COMPUTER HARDWARE AND OPERATING SYSTEMS

2018 Winter Term 1
WHO AM I?

- Jonatan Schroeder
  - Lecturer, Department of Computer Science, since 2016
  - Before that, Teaching Assistant, Research Assistant, Sessional Lecturer
  - Office: ICCS 241
About Registration

- **Current situation: as of August 30**
  - 155 registered in the course
  - 26 in waitlist
  - 34 students not registered in a tutorial (including waitlist students)

- **Waitlist: please come to class for now**

- **Tutorials**
  - No official requirement to be in a tutorial
  - If you are not registered, please come to one that fits your schedule for now
INTRODUCE YOURSELF

- Find 3 people around you, preferably people you have never met
  - Form a group

- Introduce yourself to each person in the group

- Topics of discussion
  - What area of Computer Science you are interested in?
  - Why are you taking this course? What do you want to learn?
Hiding complexity is usually good
  - All computation is mechanical
  - At the machine-level all computation is extremely complex

We work at a higher-level whenever we can
  - Design, describe, evaluate and debug
Abstractions sometimes hide too much
- performance, correctness, security are not fully covered in one layer

Good programmers must be able to dig deeper
- understanding computation at various levels of abstraction
  - including the lower layers of abstraction describing how hardware works

Extra reading on this topic on Canvas
LEARNING TO THINK ACROSS THE LAYERS

- Good mental models of multiple layers
  - CPU implementation
    - What is concurrent and what is sequential
  - Caches and the memory system
  - Disks
    - Virtual memory
    - File systems
  - Operating systems
int foo(int j, int k) {
    int i, l;
    l = j;
    for (i = 0; i < k; i++)
        l = bar(l, i);
    printf("foo(%d, %d) = %d\n", j, k, l);
    return l;
}
COURSE LEARNING GOALS

- After you complete this course, you will be able to:
  - Utilize your knowledge of the CPU and memory systems to optimize C/C++ code to make it run faster
  - Describe a possible high-level architecture for a pipelined CPU.
  - Explain the importance of, and the issues with, instruction-level parallelism and its implementation.
  - Describe the various types of memory used by modern computers, and explain how the hardware and the operating system cooperate to manage this memory.
  - Explain the issues that must be considered while designing file systems, and how file systems are managed.
  - Explain and trace how a paged virtual memory system works.
WHICH MEANS

- if you take this course seriously, you will ...
  - be better at writing, reading and debugging programs
  - be better at designing solutions to complex problems
  - be better able to understand how computers compute
  - see the importance of and issues with instruction-level parallelism
  - see the importance of and issues with system-level software
  - be a better C/C++ programmer
  - be better able to analyze and solve system-level programming problems
To Achieve This

- We will explore
  - Computer architecture
  - Storage hierarchies
  - Storage management
  - Operating systems
  - Virtual memory

- With respect to issues like:
  - persistence, synchronization and failure
  - concurrency
TENTATIVE LIST OF UNITS

- Designing a sequential CPU for the Y86-64 ISA
- Designing a pipelined CPU for the Y86-64 ISA
- Caching and the memory hierarchy
- Disks and file systems
- Operating systems (processes) and virtual memory
BASIC COURSE INFORMATION

- Pre-requisites
  - CPSC 213, CPSC 221
  - Alternative route: CPSC 210, 260 and EECE 320 instead of CPSC 221
  - I cannot waive these pre-requisites, if you don’t have them, talk to a department advisor

- Other resources
  - Canvas: pre-readings, study material, administrative information, tutorial handouts, assignment descriptions
  - Stash: assignment code check out and hand in
  - Piazza: Q&A discussion, course announcements
  - Gradescope: graded quizzes and regrade requests
CONTACT

- My office: ICCS 241
  - Generally, if my door is open feel free to ask me
- Contact me: https://www.ugrad.cs.ubc.ca/~cs313/contact
  - Avoid using my personal email for course related matters
- Office hours: listed on Canvas
- For general questions about course material or assignments, do not email directly. Instead:
  - Use Piazza
  - Ask a TA during tutorials or office hours
  - Ask me before or after class or during office hours
TEACHING ASSISTANTS

- Our teaching assistants are:
  - Siyuan He
  - Tanner Johnson
  - Mohammad Mehr Ali
  - Russell Blickhan

- Office hours: link listed on Canvas
TEXTBOOK

  - If you’ve taken CPSC 213 you should have it
  - 2nd edition should work for most of the material (references may be different)

- Other useful references listed on Canvas
CANVAS

- [https://canvas.ubc.ca/](https://canvas.ubc.ca/)
- We will use Canvas for:
  - Learning Goals and Readings
  - Assigned problems
  - Additional references
  - Assignments
  - Course policies and administrative information
- We will not use Canvas for:
  - Discussion and announcements (use Piazza instead)
  - Communication (use contact link instead)
SLIDES

- Slides will be regularly posted on Canvas
  - In Powerpoint format (with fonts embedded to minimize compatibility problems)
  - PDF if possible, but may not be updated as often if slides change
  - As a UBC student you get access to a free version of Office 365 (check link on Canvas)

- As much as possible will be posted in advance
  - Sometimes they may be modified shortly before class
Discussion board
- Do not use the discussion board on Canvas

To join:
- Go to Canvas
- In the Resources module, check the instructions
- The access code is listed in the instructions

You may use a fake name if you want – but you have to tell me what it is by sending me the details
TUTORIALS

- Tutorials start next week
- Tutorials will
  - Provide help/guidance for the assignments
  - Work on problems relevant to the course material
GRADING SCHEME

- Grading Scheme:
  - Assignments (4): 30%
  - Quizzes (5): 30%
  - Final exam: 40%

- In order to pass you must:
  - Get 50% in the final exam
  - Get 50% in the overall average of the assignments

- I reserve the right to make minor modifications to the rules above
PRACTICE THROUGH ASSIGNMENTS

- Code delivery and handing in will use Git
  - Stash: department-provided Git server, based on commercial version of BitBucket
- Your assignment must compile and work in the department computers
  - Assignments will be tested in the department’s Linux environment
- Late assignments: penalized at 33.33% per day prorated by minute
  - Assignments not accepted after 48 hours
  - Hand in time determined by last commit pushed to the Stash server
ASSIGNMENT SCHEDULE

- Assignment 1: Assembly (disassembler)
  - Due: September 28
- Assignment 2: Pipelining
  - Due: October 19
- Assignment 3: Caching
  - Due: November 9
- Assignment 4: File Systems
  - Due: November 30
EXAMINATIONS

- Biweekly quizzes:
  - Every other Wednesday starting September 19, in class
  - Each quiz is 45 minutes long

- Final exam is 2 ½ hours long
  - *Do not* make any travel arrangements for December until after the final exam date has been set
SOME RULES: USE OF LAPTOPS

- Laptops can be useful
  - Note taking, following slides, checking material, in-class activities

- However there are problems
  - Research suggests taking notes on paper provides better knowledge retention
    - http://www.brainrules.net/
  - Facebook, messaging, Web surfing, playing games, watching videos distract you and your classmates around you
SOME RULES: USE OF LAPTOPS (CONT.)

- Rule: laptops are to be used only for CPSC 313 course related work
  - Do not use it for other purposes (Facebook, Web, etc.)
  - Do not use it for course work for other courses
  - If possible, sit towards the back of the room
  - Operate on dimmest settings
  - Close laptop when not in use

- Public shaming is permitted (or comment privately to me)
Some Rules: Absences

- If you must be absent in an exam, contact me as soon as you are aware of the problem.
- You do not need to notify me of absences in days when there is no graded work.
- For assignments, there is no special consideration for expected absences.
- You may need to provide documentation:
  - Medical doctor’s note, accident report, court appearance noticed, etc.
- Failure to do so could result in a grade of zero.
- Full policy and notification instructions available on Canvas.
SOME RULES: MARKING DISAGREEMENTS

- You have 1 week from when a piece of graded work is handed back to inform me of any errors in addition, missing components or similar problems.

- If you believe something was incorrectly marked you have 1 week from when the work was handed back to bring this to my attention:
  - Provide a detailed response explaining why the answer, as written, is correct.
  - “But I meant …” is not a valid response.
  - After 1 week the mark stands.

- When submitting something for marking reconsideration, I reserve the right to review the marking of other questions for accuracy and consistency.

- Full policy and notification instructions available on Canvas.
ACADEMIC CONDUCT

- TL;DR: Don’t cheat!!
- What’s allowed:
  - Helping each other understand material and assignments
  - Exploring/discussing solutions to assignments
  - Discussing assignments with no looking at each others code or exchanging anything written (i.e. talk but don’t write)
  - Using existing public approaches to a problem – but you must properly cite the work
  - Discussing with current 313 students existing approaches to solving a problem
  - Discussing requirements
  - Discussing the merits of a proposed solution with the course instructor or TAs
ACADEMIC CONDUCT (CONT.)

- What’s not allowed
  - Submitting someone else’s work as your own
  - Accessing someone else’s answer through illicit means or by exploiting vulnerabilities
  - Having in your possession previous solutions to assignments (instructor’s or other student’s)
  - Working in a group but submitting the work individually
  - Submitting code you have handed in to another course (unless permitted)
  - Making a solution available as an aid to others, intentionally or by accident
  - Sharing code between devices or partners using public tools (e.g., public Github repo)
ACADEMIC CONDUCT (CONT.)

- If you are uncertain: ASK!!!

- Possible penalties
  - Failing grade in the work in question or in the course in general
  - Suspension from the University
  - Reprimand with letter in student’s file
  - Notation in student’s permanent record

- More information on Canvas
THINGS TO DO

▪ Register on Canvas
  ▪ Read all Policies and become aware of the resources

▪ Register on Piazza

▪ Install an SSH client on your laptop
  ▪ Windows: https://my.cs.ubc.ca/docs/free-terminal-emulation-software-xmanager
  ▪ Linux/Mac: make sure SSH package is installed

▪ Install Git on your laptop
  ▪ If using Eclipse or IntelliJ, there are plugins for Git