Outline

- Today:
  - Administrivia
  - Introduction to 225, Stacks

- Course website:
  - All slides available before lecture on website
  - Acknowledgement

John Edgar
CMPT 225 Topics

- Data Structures
- Algorithms
- Programming
CMPT 225 Topics

- Data Structures and Abstract Data Types
  - Stacks
  - Queues
  - Priority Queues
  - Trees
  - Graphs
  - Hash Tables
- Algorithms
- Programming
CMPT 225 Topics

- Data Structures
- Algorithms
  - Tools – Recursion
  - Efficiency – O Notation
  - Algorithms to support data structures
  - Sorting
- Programming
CMPT 225 Topics

- Data Structures
- Algorithms
- Programming
  - C++ coding skills
  - Memory management, pointers
Course Objectives

- Develop problem solving techniques
  - To take a problem statement
  - And develop a computer program to solve the problem
- A solution consists of two components
  - Algorithms
  - Data storage
Course Focus

- Problem solving
  - Use abstraction to design solutions
  - Design modular programs
  - Use recursion as a problem-solving strategy
- Provide tools for the management of data
  - Identify abstract data types (ADTs)
  - Examine applications that use the ADTs
  - Construct implementations of the ADTs
People

- Instructor: Greg Mori
  - Office hours:
    - Tues 14:30-15:30, Wed 15:30-16:30 in TASC1 8007
- TAs: Hossein, Jinling, Yasaman
  - Office hours:
    - Thurs 11:00-13:00, Fri 11:00-12:00 in CSIL
Prerequisites

- CMPT 125/126/128 and MACM 101
  - Familiarity with algorithms
    - E.g. searching, sorting
    - Some running time analysis, recursion
  - Knowledge/use of fundamental data types
    - E.g. numbers, characters, booleans
  - Basic programming skills
    - E.g. Loops, ifs, I/O
  - Java or C++
Prerequisite Quiz

- What does this do?

```c
int nums[] = {4, 20, 19, 3, -1, 42, 22, 18};
for (int i=0; i<8; i++) {
    for (int j=i+1; j<8; j++) {
        if (nums[i] < nums[j]) {
            int t = nums[j];
            nums[j] = nums[i];
            nums[i] = t;
        }
    }
}
```
Schedule

- See syllabus for list of topics
Assessment

- Assignments – 12%
- Labs – 8%
- Written midterm exam in class – 15%
- Programming exams in CSIL – 20%
- Final exam – 45%

- Grades will be recorded in CourSys
Assignments

- Individual programming assignments
  - Dates on syllabus

- Note late policy
  - 4 “grace days” for the trimester

- Academic Honesty (cheating)
  - We’re very serious about it
    - 17 cheating cases last time I taught this course
  - See syllabus
Labs

- Tuesdays in CSIL ASB 9840
  - NONE TOMORROW

- Instruction from TAs and fellow students
- Released in advance (Lab 0 available now)
- 1% each
  - ½ credit if late, but DON’T fall behind
Software

- All programming assignments in C++
- Labs will use
  - Linux, command-line interface
  - Programming midterm will be in lab

- If you’ve never used Linux, go to the lab this week (starting today is a good idea)
  - Do Lab 0
  - Do a Linux command-line interface tutorial
Exams

- Midterm exam in class March 11
- Programming exams in CSIL in final 2 lab slots
- Final exam Saturday April 20 8:30-11:30 location TBA
  - Comprehensive, covers whole course
- All exams are closed-book
- See syllabus regarding medical factors
No “required” books
- I.e. no homework questions, etc. from books

Reading alternative descriptions before/after lecture highly recommended

C++ language help
- Reference books, online resources listed
And on to the course...