CSc 103 Final Exam Review

General Remarks

Format. The exam will consist of short answer questions, maybe something of the "what is the output of the following code?" variety, and quite a few questions that ask you to write a small segment of code or implement some function.

Content. The topics are outlined below. Note that while the focus will be more on the material since the midterm, the final will be *cumulative*. (The material we covered before the midterm is so fundamental that it is hard to avoid testing you on it, one way or another.)

How to study. The best way to study is probably to make sure you know how to do all of the homework assignments and the "TODO" items from the notes. So if you've been following along all semester as you should, there's really not much to study! Other things to consider are the list of practice problems I gave you way back at the beginning of the term, and browsing through the relevant parts of Professor Li's notes.

List of Topics

Here are some of the main things covered since the midterm. But remember – the final is cumulative.

- STL classes.
  You should know how to use all of the following:
  - `map` was the main new thing, but don't forget about:
  - `vector`
  - `string`
  - `set`
- Recursion

  - Know how to trace recursive programs
  - Know how to write recursive programs:
    * Find self-similarity in the problem
    * Solve base case
    * Given solutions to sub-problems, construct the solution to the next increment

- Pointers, dynamic memory, dynamic classes.
  - You should be very comfortable with the concept of pointer variables, how to use them, etc.
  - Understand the equivalence between arrays and pointers (of course every array is a pointer, but every valid pointer could also be thought of as an array, perhaps of size 1...).
  - You should know the difference between dynamically allocated memory and statically allocated memory.
  - You should know how to dynamically allocate memory (that is, know what the `new` and `delete` operators do and of course how to use them).
  - Understand constructors and destructors.
  - Know what a copy constructor is, and why a class with dynamic memory needs one. Ditto for assignment operator.
  - Know a bit about operator overloading for classes.
  - Know how to write some basic functions for a linked list.