CSCI1320 Final Review Sheet

The final for this class will be a bit longer than the midterm. It will focus primarily on the material from the second half of the semester, though material from the first half will certainly be present on it and one or two questions will be specifically on topics from before the midterm. The final will have 11 regular questions and one extra credit question. Some of the 11 regular questions will have multiple parts and they are weighted differently. The style of questions will be similar to what you have seen before with short answer, code writing, and tracing of code.

Topics:

Functions

You need to understand the concept of breaking problems up into pieces. Know what functions are and how to write them. Know about function literals and how to write them. Have some understanding of higher order functions.

Conditional statements

Know the syntax of if and match expressions and what they can do and are used for. Boolean expressions, comparisons, and building complex expressions with logic operators. Know how to combine conditionals with functions to produce basic recursion.

Collections

Know the basics of arrays, lists, sets, and maps including how to construct them and work with them. Understand the basic methods available on collections. Understand the higher order methods available on collections. You should understand how to use 2-D array (arrays of arrays).

Repetition

Know the fundamental aspects of loops and the main types of loops. While, for, and do/while loop syntax and usage. Know how you can use recursion to produce repetition.

Files

You need to know how to use files in code as well as what they are good for. If you have to write code involving files you don't have to have the function call syntax perfect, but it needs to be close enough that I can understand. It should definitely include the right names of the functions.

Case Classes

Know how to declare and use case classes.

Mutable and Immutable

Understand the difference between mutable and immutable data. Be especially aware of what impact this difference has on data passed into functions.

Sorting

You need to understand how each of the three $O(n^2)$ sorting algorithms works. You should be able to write the code for them using loops or recursion.

Searching

Know how sequential and binary searches work. You need to know when each one can or should be used and be able to describe why the binary search is faster.

Recursion

Know how to write and trace recursive functions. You should also understand when recursive functions can/should be used to aid in problem solving. How do they compare to loops and when are they superior to loops?

For all language features, you need to know how to use them both in writing code and be able to trace code that involves them.