CSCI 104L Final Review

ALGORITHMS:
- Recursion: Programming, and Analysis
- Sorting algorithms: runtimes, loop invariants, stability
- Search algorithms: Linear, Binary, Interpolation
- Graph search: BFS, DFS, Dijkstra, A*
- Hash functions

GRAPHS AND TREES:
- Undirected/directed/weighted
- Adjacency Lists, Adjacency Matrices
- Complete trees
- Binary Search Trees
- Tree traversals
- Backtracking

DATA STRUCTURES:
- for all of them: the functions they provide, how and where to use them, how they are implemented, how fast the operations are.
- Linked Lists
- Arrays and Lists
- Heaps
- AVL Trees
- Hash Table/Bloom Filters
- Tries/Compressed Tries
- Suffix Trees
- Splay Trees
- 2-3 Trees
- ADTs: Lists, Queues, Stacks, Maps, Sets, Priority Queues

RUNTIME ANALYSIS:
- Definition of worst-case time
- How to provide upper and lower bounds
- How to perform calculations with big-O/Omega/Theta
- Setting up sums for loops, and recurrences for recursion
- Basic sums: arithmetic, geometric, harmonic
- Amortized runtime (averaging runtime over multiple calls)

C++ PROGRAMMING:
- object-oriented design
- Abstraction and Encapsulation (packing common-purpose elements into the same class).
- structs and classes: public, protected, private
- calling member functions
- constructors and destructors, including shallow and deep copies
- pointers, passing by value and reference
- operator overloading and friend access
- inheritance and polymorphism in C++, including virtual functions, abstract classes, and scoping (Base-Class::function)
- the const keyword, and how and why to use it.
- exceptions
- iterators
- templates