

**CSCI 2110**  
**Data Structures and Algorithms**  
**TOPICS FOR THE FINAL EXAM**

**Module 1: Object-oriented programming concepts**

- Quick review of encapsulation and inheritance
- Polymorphism
- Generics

**Module 2: Algorithm Time Complexity**

- What is algorithm complexity?
- Big O notation
- Deriving Big O given a function.
- Problems with Big O.
- Examples of algorithms with various Big Os.
- Best case, worst case and average case complexity

**Module 3: Unordered Lists**

- Definition and examples
- Generic set of operations and the big picture of implementation
- Linked List Implementation (methods in the LinkedList class)
- Understand List methods and how to use them
- Complexities of operations
- Best case, worst case and average case search; smart search method

**Module 4: Ordered Lists**

- Definition and examples
- Binary search algorithm and its complexity
- Understand OrderedList methods and how to use them
- Complexities of operations
- Merging ordered lists – two finger walking algorithm

**Module 5: Recursion**

- Quick review of recursion and examples

**Module 6: Binary Trees**

- Definition
- Binary Tree Terminology
- Strictly binary and complete binary trees
- Traversals
- Recursive definitions for a binary tree
- Binary Tree Class
- Application: Huffman coding

**Module 7: Binary Search Trees**

- Definition and Examples
- Operations – Search, Insert and Delete
- Binary Search Tree Class
- Complexity of Binary Search

**Module 8: AVL Trees**

- Motivation and Definition
- AVL tree examples
- AVL tree rotations with examples

### **Module 9: Heaps**

- Motivation, definition and properties
- Operations – Search, Insert and Delete
- Heap Class
- Heap Applications – Sorting and CPU Process scheduling
- Updatable Heaps

### **Module 10: Hashing**

- Hashing concept, hash tables and hash functions
- Hash Collision
- Separate chaining
- Linear and Quadratic probing
- Hash Functions
- Strings as keys
- Load Factor
- Rehashing
- HashMaps, TreeMap and LinkedHashMaps

### **Module 12: Graphs**

- Graph Definitions and Terminology, Applications of graphs
- Graph Types
  - Directed vs. Undirected
  - Weighted vs. Unweighted
  - Strongly connected vs. weakly connected)
- Representation of an edge
- Degree of a vertex, Indegree vs. Outdegree
- Representation of graphs
  - Adjacency Matrix Representation
  - Linked List Representation
- Graph Traversals – breadth first, depth first
- Topological Sort
- Shortest Path Algorithm

**Note:** Reference Sheets will be supplied for the following:

1. Formulas for basic log operations, sum of series and exponentiation.
2. Method headers and meanings for important classes such as Node, LinkedList, Unordered, Ordered, Binary Trees, Binary Search Trees, Heaps, Hash Tables. However, please read the codes and understand how each method works.
3. AVL tree rotations – diagrams for general cases. Know how the rotation works in examples.