

CSCI 230 – Data Structures and Algorithms

Fall 2019 Midterm Study Guide

1. Intro to Data Structures and Algorithms
 - a. What is a data type?
 - b. What is an Abstract Data Type?
 - c. How do we use data types or ADTs?
2. Object Oriented Programming
 - a. What is inheritance? Give an example.
 - b. What is encapsulation? What is its difference with information hiding?
 - c. What is polymorphism? Give an example
3. Algorithm Analysis
 - a. Difference between algorithm and program
 - b. Calculate the upper bound for the best, worst, average case of an algorithm.
 - c. Calculate the time / number of operations $T(n)$ for an algorithm.
 - d. How much faster is an algorithm with $O(f(n))$ when we run it in a faster computer?
 - e. Calculate the runtime big-Theta for an algorithm.
 - f. Use the definition for Big-O, Big-Omega, Big-Theta to prove that a specific upper, lower, or runtime bound respectively, works.
4. List
 - a. Reproduce all basic methods of the List ADT interface with ArrayList or LinkedList
 - b. Any variation of standard operations to the list:
 - i. Insert to a specific index / position
 - ii. Remove from a specific index / position
 - iii. Find duplicates
 - iv. Find element e
 - v. Print reverse
 - c. Analysis of list operations
 - i. Time
 - ii. Memory
5. Stack
 - a. Use a stack to implement a freelist.
 - b. ArrayStack implementation
 - c. LinkedStack implementation
 - d. Analysis of stack operations
 - i. Time
 - ii. Memory
6. Queue
 - a. Cyclic array implementation of ArrayQueue
 - b. Linked Queue implementation
 - c. Analysis of queue operations
 - i. Time
 - ii. Memory

7. Recursion

- a. Problem solving with recursion.
 - i. Find the base case.
 - ii. Find the recursion.
 - iii. Write the whole program.
- b. Analysis of time complexity of recursive problems.