# CS 284: Endterm (50 Minutes) – Fall 2017

December 8, 2017

Student Name:

Honor Pledge:

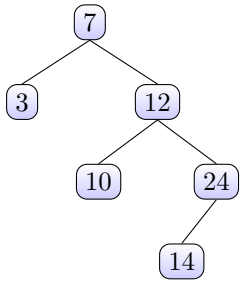
Grade sheet:

| | |
|---|---|
| Problem 1 (20 points) | |
| Problem 2.1 (10 points) | |
| Problem 2.2 (10 points) | |
| Problem 3 (20 points) | |
| Problem 4 (20 points) | |
| Problem 5 (20 points) | |

# Problems

## Exercise 1 (*Trees*)

Implement a method `public ArrayList<ArrayList<Integer>> paths(Node<E> localroot)` that returns the list of all the paths in the tree rooted at `localroot`. A *path* is a sequence of 0 or 1 that denotes a path in the tree from the root to a node. For example, the set of paths of the tree below is `[[], [0], [1], [1, 0], [1, 1], [1, 1, 0]]`.



## Exercise 2 (*Heaps*)

1. Build the max heap that would result from inserting the following elements in the order in which they are presented. Show the intermediate heaps resulting from each individual insertion.

   15, 25, 47, 33, 55, 82, 90, 10, 18

2. Show the heap resulting from performing a deletion.

## Exercise 3 (*Sorting*)

Sort the following list using insertion sort

72, 35, 18, 22, 43, 12, 52, 21, 12

Exhibit the list resulting from each pass.

## Exercise 4 (*Sorting*)

Explain why the modified *partitioning* algorithm in quicksort that, given an array `a` of size `n`, selects the pivot to be the middle element in the set {`a[0]`, `a[n div 2]`, `a[n]`}, is better than just taking the pivot to be `a[0]`.

## Exercise 5 (*Hashing*)

Insert the items in the following table

| Key | Index |
|---|---|
| A | 1 |
| B | 4 |
| C | 5 |
| D | 10 |
| E | 5 |
| F | 2 |
| G | 1 |
| H | 10 |
| I | 1 |

into the hash table below. Resolve collisions using linear probing.

| | |
|---|---|
| 0 | |
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |
| 9 | |
| 10 | |