

CSC 212: Data Structures and Abstractions
Spring 2018
University of Rhode Island
Weekly Problem Set #11

Due Thursday 4/19 before class. Please turn in neat, and organized, answers hand-written on standard-sized paper **without any fringe**. At the top of each sheet you hand in, please write your name, and ID. The only library you're allowed to use in your answers is `iostream`.

1 Binary Search Trees

1. Draw a binary search tree after the following operations steps:
 - (a) Insert: [10, 5, 12, 8, 19, 6, 2, 11, 15, 9, 7]
 - (b) Remove: [7, 12, 8, 10]
2. Write a function to delete binary trees. Be sure to remove nodes in the proper order, so that none get orphaned.
3. Write a recursive function that, given a binary tree, returns a tree of the same shape, where every node's value has been set to it's own depth in the tree. You should use two parameters in your call: a pointer to a node, and current depth.
4. Briefly explain the difference between in-order, post-order, and pre-order traversals.

2 2-3 Trees

1. Draw a 2-3 tree after the following operations steps:
 - (a) Insert: [10, 5, 12, 8, 19, 6, 2, 11, 15, 9, 7]
 - (b) Remove: [7, 12, 8, 10, 9]
2. Write a search algorithm for a 2-3 tree.
3. Explain the tradeoffs between binary search trees, and 2-3 trees. Justify your answer, citing specific examples where you would use either type of tree.

3 Optional Problems

1. Implement a binary search tree with all of the following methods: constructor, destructor, insert, search, remove.