Problem 1

(Adapted from Weiss 4.9 and 4.19)
In this problem you will practice insertion into binary search trees and AVL trees.

A (2 pts) Show how to insert 3, 1, 4, 6, 9, 2, 5, and 7 into an initially empty binary search tree. (Show each step.)

B (3 pts) Show how to delete the root from the binary search tree you created. (Show all work.)

C (5 pts) Show how to insert 2, 1, 4, 5, 9, 3, 6, 7 into an initially empty AVL tree. (Show each step, including rebalancing.)

Problem 2

In this problem you will practice insertion and deletion in binary heaps (default min heap).

A (7 pts) Show how to insert 10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, and 2 into an initially empty binary heap. Insert each value, one at a time (not with buildHeap), and show each of the 15 steps as separate trees (pictorially with nodes and edges). For only the step of adding the 3, show the initial array representation and each step of the percolate up until the 3 is in the right place.

B (7 pts) Show the results of two consecutive deleteMin operations on the heap above (show each). For only the first deleteMin, show the initial array representation and each step of the percolate down until the operation is complete.

Problem 3

(Adapted from Weiss 8.1)
In this problem you will practice working with the union-find algorithms and up-tree data struc-
ture. You are given 17 individual sets numbered 0 through 16. Show the results of the following sequence of instructions (show each step as a tree):

union(1,2), union(3,4), union(3,5), union(1,7),
union(3,6), union(8,9), union(1,8), union(3,10),
union(3,11), union(3,12), union(3,13), union(14,15),
union(14,16), union(1,3), union(1,14)

when unions are:

A (7 pts) Performed arbitrarily by making the second argument a child of the first argument.

B (7 pts) Performed by size.