Eduard Klimenko
TCSS 342 Data Structures
Assignment 4

1. A.
Since each node of a BST contains a left child that is smaller and a right child that is larger, it is logical to conclude that the root should be in the middle of the set (from 1 to n). After inserting the middle element we split the set into two, one left of the middle and one to the right. Working with the left set we repeat the same steps, inserting the middle element and partitioning the set. After there are no more elements we do the same thing with the right set. This will ensure minimum height.
B.
With minimum height, the operations find, insert, and remove will all be done in worst case $O(log(n))$.
C.
To ensure maximum height, elements should be inserted in increasing order (from 1 to n). This will always put the next element as the right child, making the tree act as a single linked list.
D.
The worst time complexity of find, insert, and remove operations are done in $O(n)$.

4. Binary Search Tree

Number of Nodes	Average Height	Minimum Height	Constant (average/minimum)
10	5.75	2.459	2.339
100	13.23	5.658	2.338
1,000	22	8.967	2.453
2,000	24.063	9.967	2.414
5,000	27.706	11.288	2.454
10,000	31.059	12.288	2.528
100,000	40.294	15.610	2.581
1,000,000	50.143	18.932	2.649

The relationship between constants for each n is easy to see. The average height of a BST is around 2.5 times the minimum height possible for n number of nodes.