Assignment 2 Marking Scheme

Robin Dawes

October 16, 2021

Implementation/Code (65%):

Implement Binary Tree Class with Specified Functionality (40%)

- a constructor that returns a reference to a new, empty Binary Search Tree 5%
- an instance function **insert(x)** that adds integer x to the tree (duplicates allowed) 10%
- an instance function **search(x)** that returns a reference to a vertex in the tree that contains x, or a null (nil, none) reference if x is not in the tree 5%
- an instance function **remove(x)** that removes integer x from the tree. If the tree contains multiple instances of x, only one instance of x is removed. If the tree contains no instances of x, the tree is unchanged 10%
- an instance function **height()** that returns the number of levels in the tree 10%

Experiment (15%):

Correctly implement the experiment as per the assignment.

Programming Style (10%):

Code should be well organized and documented.

Report (35 marks):

Presentation of Results (15%):

Present results in a table or graph showing the k value, the t value, and the percentage of tall trees for each of the specified values of n

Research Questions (20%):

1. As n increases, does the average height appear to grow at a logarithmic rate, a linear rate, or something else? 10%
2. Does the percentage of trees with height $\geq \frac{n}{2}$ grow, shrink, or remain fairly constant as n increases? 10%