Homework 4

Due 10/03/2024

September 26, 2024

1. Sketch the contents of an AVL tree after adding all of the values 1–10 in sorted order to an empty AVL tree.

2. Sketch an AVL tree that contains values 1–n (where n is the number of nodes in your tree) in which removing one node results in *exactly* three rotations: right, then left, then left again. Clearly indicate which node in your tree has this property.

3. List all coefficients c such that the hash function $h(x) = cx \bmod capacity$ can potentially return any index when $capacity = 24$. Only list coefficients in the range 0 to 23 (inclusive).

4. Draw the contents of a hash table with capacity 20 using open addressing with linear probing and the hash function $h(x) = 7x + 5 \bmod 20$ after inserting the values 1, 5, 11, 18, 3, and 8 into an empty hash table.

5. Describe an efficient implementation of a Set ADT that combines the advantages of a hash table and a balanced binary search tree. In particular, the search function for your data structure should have $O(\lg n)$ worst case complexity and $\Theta(1)$ expected complexity, and it should not be limited to values in a small range. Your description should include:

   (a) how you are storing data in memory,

   (b) how you search for elements, and

   (c) how you insert new elements into the data structure.

   Hint: your description should relate to existing data structures. Do not reproduce pseudocode for any of the operations for any structure discussed in class. Instead, describe how you would modify these operations.