Module 8 Problems

1. [6pts] Draw the 2 - 3 tree that results from inserting

      M E D I U H A R Q U T O N

in the given order into an empty 2 - 3 tree.

2. [6pts] Draw the 2 - 3 tree that results from inserting

      Y L P M Z H C R A E S

in that order for an initially empty 2 - 3 tree.

3. [6pts] Draw the Left Leaning Red/Black that inserts the problem 1 sequence into the empty tree.

      M E D I U H A R Q U T O N

(sequence repeated for convenience)

4. [6pts] Draw the Left Leaning Red/Black that inserts the problem 2 sequence into the empty tree.

      Y L P M Z H C R A E S

(sequence repeated for convenience)

5. [12pts] Give Python code for the delete operation for left-leaning red-black trees.
Justify your answer.

6. [12pts] How do you find the minimum element in a red/black tree?
Give Python code code, correctness justification, and complexity analysis for your algorithm

7. [14pts] How would you find the median value in a LLRB with numbers as node values?
Give Python code, a correctness justification, and a complexity analysis for your algorithm
Hint: You may want to augment the LLRB nodes with metadata. If you decide to do so, part of your
correctness justification will be to indicate how the metadata can be accurately maintained during insert and
delete operations.

8.1 [3pts] Using h(i) = (2*i + 5) mod 11, insert the following sequence into the empty 11-item hash table:

       12, 14, 34, 88, 23, 94, 11, 39, 20, 16, 5

Show the final hash table with collision lists.

8.2 [3pts] Using the same hash function h, insert the 8.1 sequence into an empty 11-item hash table,
but use linear probes to handle collisions.

9. [12pts] Given 2 sets of numbers H, J, |H| = |J| = n, and a value K, find an algorithm with expected time O(n) to determine if there are 2 numbers h ∈ H, j ∈ J s.t. h + j = K.

For example:
     Let H = {4, 5, 7, -1, -9, 22}.
     Let J = {0, -5, 10, 44, 100, -12}.

    In this case, |H| = |J| = 6.

    Let K = 91. Your algorithm should answer 'yes', giving -9 (In H) + 100 (in K).
    Let K = 92. Your algorithm should answer 'no' .

10. [8pts] For your problem 9 algorithm:
    a) What is the worst-case complexity of your algorithm?
    b) When does the worst case happen?

11. [12pts] Given a single array H, |H| = n, and a value K, give an expected time O(n) algorithm to determine if there are 2 elements of H, h1, h2 s.t. h1+h2 = K. **Caveat**: The algorithm is similar to, but not identical with problem 9.