Test 2

First Name: ______________________________

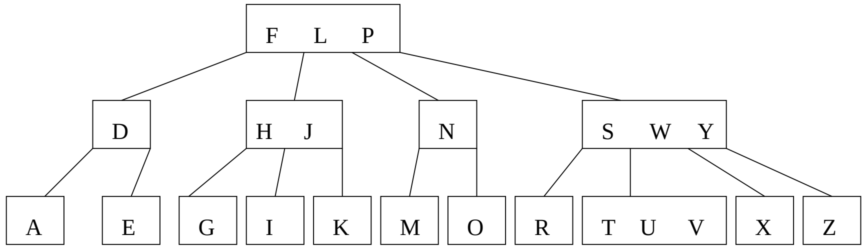
Last Name: ______________________________

Student Number: ______________________

This test lasts 75 minutes.
Aids allowed: one 8.5×11 *inch piece of paper with handwritten notes.*
Make sure your test has 6 pages, including this cover page.
Answer in the space provided. (If you need more space, use page 6 and indicate clearly that your answer is continued there.)
Write legibly.

Question 1	/3
Question 2	/2
Question 3	/2
Question 4	/7
Question 5	/5
Question 6	/4
Total	/23

[3] **1.** Show the result of inserting the key Q into the following B-tree.



[2] **2.** Let x be a node in a red-black tree T. Suppose a new key is inserted into T. How can the depth of node x change as a result of this insertion? (Can it increase or decrease? By how much?)

[2] **3.** Recall the forest data structure for the disjoint sets (union-find) abstract data type which uses union by rank and path compression. Briefly explain why the union by rank heuristic is used instead of union by height.

[7] **4.** Describe how to implement the SUCCESSOR(k, x) operation in a B-tree. The arguments to the function are a key k stored in the tree and a pointer x to the node that contains k. It should return the smallest key in the data structure that is larger than k. (Assume all keys in the tree are distinct and that k is not the largest key in the tree.)

What is the worst-case running time of your algorithm. State your answer in terms of n, the number of keys in the tree, and t, the minimum number of children an internal node can have in the B-tree. Use Θ notation. Briefly justify your answer.

[5] **5.** Consider an abstract data type that stores a set of distinct integers and supports 3 operations:

- INSERT(i) adds the integer i to the set (with the precondition that i must not already be in the set when INSERT(i) is called),
- DELETE(i) removes i from the set (with the precondition that i is in the set when DELETE(i) is called), and
- EXTRACT-AVERAGE removes the element of the set that is closest to the average of the set and returns it. (If two elements are equally close to the average, it should remove the smaller of the two. If the set is empty it should return 0.)

For example, if the dictionary contains $\{1, 7, 3, 12\}$ the average is $\frac{1+7+3+12}{4} = 5.75$, so an EXTRACT-AVERAGE would remove 7 from the set and return it.

Explain how to use a red-black tree to implement this abstract data type so that the worst-case running time of each operation is $O(\log n)$, where n is the number of integers in the set. You may augment the red-black tree with extra information if you want (but explain clearly any changes this would cause). You may call any of the red-black tree routines we discussed in class.

[4] **6.** Assume we have *any* implementation of the abstract data type defined in Question 5 where the worst-case time for each operation is $O(\log n)$ when the set has n elements in it.

Give a good upper bound on the *total* time to do a sequence of m operations, of which m_1 are INSERTS, m_2 are DELETES and m_3 are EXTRACT-AVERAGES. Assume the data structure is initially empty. Justify your answer carefully.

(Hint: you do not need to know the answer to Question 5 in order to answer this question.)

This nearly blank page is just for additional workspace, if you need it.