Homework 4 Problems  
CompSci 261P—Fall, 2019—Dillencourt

These problems are recommended. Do not turn them in. Solutions will be posted (except for the first one.)

1. (Note: the purpose of this problem is to help you make sure that you understand the mechanics of the various data structures listed below. If you are convinced you understand how they work you can skip this problem, but be aware that you might get some version of this problem on the midterm and/or the final exam.)

For each of the following data structures, construct some sample data structures (either by inserting items one at a time or by starting with a populated instance of the structure.) Then work through the various operations.

(a) Standard binary trees: search, delete
(b) B-trees: search, insert, delete
(c) Treap: insert, delete
(d) Splay tree: insert, search, delete. Make sure you know how to splay and when/where to splay.
(e) Suffix tree. Make sure you know how to build one from a string and how to use it to find a substring within the string. You should be familiar with and able to work with all three variants:
   - prefix tree/uncompressed trie
   - Patricia trie/compressed trie
   - compacted Patricia trie
(f) Range tree: Make sure you know how to perform range queries, and how to update the data structure on insertions, deletions, and rotations.

2. In a binary search tree formed by inserting \( n > 2 \) different values in a random permutation without rebalancing, what is the probability that the node for the \( k \)th smallest value has exactly one child node (as a function of \( n \) and \( k \))? (Hint: If \( x[k] \) is the \( k \)th smallest key, think about the order in which the keys \( x[k], x[k-1], \) and \( x[k+1] \) are inserted, with appropriate attention to special cases.)

3. Suppose we want to store some items in a B-tree, and we want each B-tree node to fit on a page. Suppose each data item (key plus associate data) takes up eight bytes and each child pointer takes up four bytes. If a page is 512 bytes, what should be the order of the B-tree?

4. Suppose we have an initially empty B-tree of order 101 (i.e., a (51,101)-tree). Suppose we add integer keys to it in the order 1, 2, 3, . . . . Which key is the one whose insertion first causes the root to be at height 3? (A node with only null children is at height 0.)

5. Suppose that \( T \) is a splay tree, on three items with keys 1, 2, and 3, and that all three of the keys have already been searched for since the most recent time the set of keys in \( T \) changed. Based on this information, how many different shapes might \( T \) possibly have? Draw them all.
6. (a) Find four strings whose compressed trie forms a complete binary tree with four leaves.
   (b) Is it possible for the suffix tree of a string to form a complete binary tree with four leaves? If
       yes, provide an example; if no, explain why not.

7. Let $n$ be any positive integer, and let string $s$ (including its string termination symbol $\$$) have $n$
   symbols in it: $n - 1$ occurrences of the symbol "a" followed by the end-of-string symbol $\$$. For
   instance, for $n=5$ the string $s$ would be $\text{aaaa}\$.
   (a) How many leaves does the suffix tree of $s$ have, as a function of $n$?
   (b) How many non-leaf nodes does it have?

8. Suppose we are given the following 16 pairs:
   (1,7), (4,17), (9,29), (16,43), (25,37), (36,53), (49,13), (64,5), (81,23), (100,2),
   (121,3), (144,11), (169,41), (196,47), (225,31), (256,19).
   (a) Construct a balanced binary range tree for these elements, where the elements are sorted by
       their $x$ coordinate and the operation to be computed is the sum.
   (b) Describe how the range tree you constructed in (a) would be used to compute the sum of the $y$
       coordinates of all points whose $x$-coordinates are in the range [90,175], using the double binary
       search algorithm discussed in class. State the search path, which children of nodes on the search
       path are excluded from the sum without searching their children, and which children of nodes
       on the search path are included in the sum without searching their children.
   (c) Suppose we insert a new pair (110, 59) in the range tree, without doing any rotations. Show the
       updated range tree and carefully describe how the values stored in the tree nodes are updated.

9. The standard deviation of a collection of $n$ data values $\{x_i\}$ calculated by the formula
   $$s = \sqrt{\frac{n \sum x_i^2 - (\sum x_i)^2}{n \cdot (n-1)}}$$
   (a) Give an example that shows that the standard deviation is not decomposable.
   (b) Describe a decomposable range querying problem such that the standard deviation of the data
       values within a query range can be computed in constant time from the answer to your query.
       (We went through a similar example for the average in the lecture.)