T-301-REIR, REIKNIRIT/ALGORITHMS
HAUST/FALL 2019
D4 - SYMBOL TABLES (AND SORTING)

Problems 4 through 7 are of the problem solving type. You are to give an efficient algorithmic method that solves the given problem. Give answers as a clear but *succinct* text description (few lines each). Grade is given for serious effort.

If not specified, you should assume that input numbers can have a very large range.

Problem 1. *What is the most frequent word in "A tale of two cities" (tale.txt) of length at least 7 that starts with the same letter as your first name? How frequent is it? [Hint: Consider* `FrequencyCounter.java`*]*

Problem 2. *(Problem 3.2.4) Suppose that a certain BST has keys that are integers between 1 and 10, and we search for 5. Which sequence below cannot be the sequence of keys examined?*

(1) *10, 9, 8, 7, 6, 5*
(2) *4, 10, 8, 6, 5*
(3) *1, 10, 2, 9, 3, 8, 4, 7, 6, 5*
(4) *2, 7, 3, 8, 4, 5*
(5) *1, 2, 10, 4, 8, 5*

Problem 3. *Write the constructor BST(double a[]), that converts the unsorted array of numbers into a corresponding (properly ordered) binary search tree. The tree should be of minimum height, and the method efficient. You may use support functions.*

Problem 4. *Josie needs a data structure that can handle the following operations:* `push, pop, contains, remove`, where `contains(Item x)` answers whether a given item is in the data structure and `remove(Item x)` removes the given item from the data structure. How can this be implemented efficiently?

Problem 5. *We are given k sorted arrays, with a total of N (distinct) elements, where $k < N$. How can we output all the numbers in sorted order, using time $O(N \log k)$?*

Problem 6. *Given a collection of intervals (on the real line) and a real value x, a stabbing count query is the number of intervals that contain x. Design a data structure that supports interval insertions intermixed with stabbing count queries, in logarithmic time per operation.*

Problem 7. *Give a sequence of N insertions into a red-black tree that lead to the tree having height $\sim 2 \log N$. (You may assume $N = 2^k - 2$, for some k).*

Date: September 27, 2019.

Problem 8. *Suppose that the keys A through G, with the hash keys given below, are inserted in some order into an initially empty table of size 7 using linear probing (M=7, no resizing).*

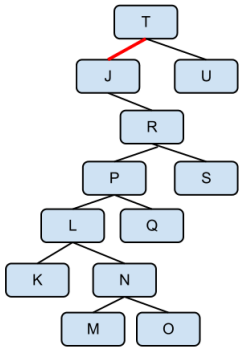
key	A	B	C	D	E	F	G
hash	2	0	5	4	4	4	2

Which of the following (more than one might apply) could not possibly result from inserting these keys? Explain briefly.

(1) B E A G D F C
(2) C F A G D E B
(3) F B G A E C D
(4) F C B G A D E

Class Problems. These problems will be discussed in the exercise sections. Please attempt all these problems before that class, as this is essential to understanding the fundamentals.

Problem 9. *(Problem 3.3.16) Show the result of inserting the letter n into the red-black BST shown on page 450 (only the search path is shown, and you need to include only these nodes in your answer).*



Problem 10. *(Problem 3.3.5, shortened) Draw all structurally different 2-3 trees for N = 7, 8, and 9. (The trees for N from 1 to 6 are shown on p. 449)*

School of Computer Science, Reykjavik University, Menntavegi 1, 101 Reykjavik

Email address: mmh@ru.is