9. Problem set on

Algorithms, Data Structures and Data Abstraction WS 18/19

Wolfgang Mulzer

Due on 21. December 2018, 10:00, in the respective TA's mailbox

Problem 1 Expected Path Length in a Binary Search Tree *10 Points*

Let n be a natural number. The *randomly constructed binary search tree* with n nodes is obtained by inserting the keys from $\{1, \dots, n\}$ in random order into a binary search tree that is initially empty (without rebalancing). Each of the $n!$ possible insertion orders is equally likely.

(a) Show that the probability distribution on the set of binary search trees with n nodes that we obtain through random construction differs from the uniform distribution on this set.

Hint: Consider the case $n = 3$.

Let T be a binary search tree and x a node in T. Let $d(x, T)$ denote the depth of x in T. The *internal path length* $P(T)$ of T is defined as

$$P(T) = \sum_{x \in T} d(x, T).$$

We will show that $\mathbf{E}_T[P(T)] = O(n \log n)$, for a randomly constructed binary search tree T with n nodes.

(b) Let $i, j \in \{1, \dots, n\}$ with $i < j$. Let X_{ij} be the indicator random variable for the event that i is an ancestor of j or that j is an ancestor of i. Show that

$$P(T) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}.$$

(c) Let $i, j \in \{1, \dots, n\}$ with $i < j$, and let X_{ij} as before. Show that

$$\mathbf{E}[X_{ij}] = \frac{2}{j - i + 1}.$$

(d) Show that

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{2}{j - i + 1} = \sum_{i=2}^{n} \sum_{j=2}^{i} \frac{2}{j}.$$

(e) Use the fact that $\sum_{j=2}^{i} \frac{1}{j} \leq \ln i$, for all $i \geq 2$, to get $\mathbf{E}_T[P(T)] = O(n \log n)$.

Problem 2 Red-Black Trees and $(2,4)$-Trees *10 Points*

A *red-black tree* is a binary search tree in which every node has a color: red or black. The following rules apply: (i) the root is black; (ii) the $\perp$-nodes that indicate the empty subtrees are black; (iii) the children of every red node are black; (iv) the *black depth* of all $\perp$-nodes is the same, i.e., there is a number h such that for each $\perp$-node v, the number of black nodes on the path from v to the root is h.

(a) Show that a red-black tree with n nodes has height $O(\log n)$.

(b) Show that red-black trees and $(2,4)$-trees are equivalent. More precisely: describe a local transformation that translates groups of nodes in a red-black tree to nodes in a $(2,4)$-tree, and vice versa. Justify why your transformation fulfills the red-black tree properties and the $(2,4)$-tree properties.

Problem 3 $(2,3)$-Trees *10 Points*

(a) Insert the keys A, L, P, D, R, E, I, X, Y, Z in this order into an initially empty $(2,3)$-tree. Show the individual steps. Afterwards, delete the keys A, L, X, R.

(b) Describe schematically a $(2,3)$-tree with n keys for which there is a deletion operation that requires $\Omega(\log n)$ rebalancing operations.