Homework 8

OPTIONAL PROBLEMS

Due never, do now

"I understand why marriages break up over golf. I can't even talk about my own handicap because it's too upsetting." -*Shia Labeouf*

1 Written Problems

Problem 8.1

Moar Treaps

Prove (by strong induction) that any given collection $(k_1, p_1), \ldots, (k_n, p_n)$ of key-priority pairs, where all keys are distinct and all priorities are distinct, there is a unique treap T with n nodes, where each node contains a different key-priority pair. "Unique" means that there is only one way to arrange the treap for a given set of inputs.

Note: Strong induction works the same way as regular induction, except instead of assuming $P(k)$ and showing $P(k+1)$, you assume $P(i)$ for all $i \leq k$, and show that $P(k+1)$ follows from that.

Problem 8.2

Sorting Nodes by Depth

Given a binary search tree, design an algorithm which creates a linked list of all the nodes at each depth. For example, if you have a tree with depth D, you'll have D linked lists. Your function should take in the root of the BST (which has pointers to any child nodes it may have), and return a list of linked lists.

Problem 8.3

Rotated Array

Given a sorted array of n integers that has been rotated an unknown number of times, give an O(logn) algorithm that finds an element in the array. You may assume that the array was originally sorted in increasing order and there are no duplicates.

Problem 8.4

Rotated Array Episode II

It is given that in the array described above, the smallest integer has a value of 1, and the array contains only numerically consecutive integers. Write pseudocode to find how many times the original array was rotated.