Homework 8
EXTRA CREDIT 1

Due Friday April 17, 5 PM EST

Extra credit is worth one extra percent of your overall class grade. TAs will not answer Piazza questions or hours queries on any extra credit topics.

To submit your solutions, please make a directory ~/course/cs0160/extraCredit1 containing your solution PDF and then run the script cs0160_handin extraCredit1.

1 Written Problems

Problem 8.1
More Treaps

Prove (by strong induction) for 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.

Note further: for a reminder on Treaps, please reference Homework 8.

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.