ESO207 Assignment-3

Submission Deadline: Nov. 7, 2020 (23 hrs : 59 mins)

Maximum marks: 100

Instructions

- Only one submission per team is allowed.
- Each team should work independently and write its own code.
- As usual, use any one of the four programming languages: C, C++, Java or Python.
- Document your program properly so that it is understandable to the reader.

Q1 Consider a class of items in which each item has (at least) two attributes, a *key* and a *priority*. A treap T of such items is a bst (binary search tree) with respect to key attribute of these items. Moreover, T satisfies a min-heap like property on priority attribute. That is, for all nodes $x \in T$, if $x \neq T.root$ then $x.priority \geq parent(x).priority$. We assume that all keys and all priorities in a treap T are distinct.

(a)(marks 60) Write a program *Insert(T,x)*, to insert an item x into a treap T.

To do this first think of an algorithm for *Insert(T,x)*.

A possible algorithm for this is to first insert x into bst T, ignoring priority attribute of x. If the heap property is violated then it is restored by pushing x upward towards the root using rotations.

You may get more details in problem 13-4 (in particular, figure on page 335) of CLRS book.

(b)(marks 5) Using the procedure in (a), write a procedure *Insert1(T,k)*, where T is a treap and k a key value. *Insert1(T,k)*, guesses a random number p as priority and Inserts (k, p) into treap T.

(c)(marks 15) For testing purpose, write procedures $inorder(T)$, $preorder(T)$ which output a list of (key, priority) pairs from nodes of T listed in inorder, preorder traversals of T respectively. Also write procedure $height(T)$, which returns height of T.

(d)(marks 10) Now, starting with an empty treap, insert items $1, 2, 3, \ldots, 100$ successively into it using procedure *Insert1(T,k)* of part **(b)**. Run *height(T)* to find height of the final treap T. Repeat this five times (each time starting with an empty treap T). Print the heights of these five treaps individually and their average. Compare these heights with the scenario where we insert items $1, 2, 3, \ldots, 100$ successively into an empty (and ordinary) bst R. What do you observe?

(e)(marks 10) Repeat part **(d)** for sequence

$12, 6, 18, 3, 9, 15, 21, 2, 1, 4, 5, 7, 8, 10, 11, 13, 14, 16, 17, 19, 20, 23, 22, 24$

instead of sequence $1, 2, 3, \ldots, 100$.

———————- x–x–x ——————