ESO207: Data Structures and Algorithms

Programming Assignment *3* *Due: October 24, 2018 23:59*

This programming assignment is to help you practice with the *binary search tree* data structure. In this assignment, irrespective of the language you are using, you have to implement the BST data structure from scratch.

The problem concerns maintenance and operations on a dynamic set of intervals. You are given a set of intervals, with insertion and deletion operations. You have to implement the operations *Insert*, *Delete*, *Min*, *Max*, *LoSucc*, *HiSucc* and *IsOverlap* where, these operations are defined as follows. An interval i is defined as the pair $(i.lo, i.hi)$. For the purposes of the problem, assume that no two intervals have the same *lo* value and no two intervals have the same *hi* value. Let T denote the dynamic set of intervals.

1. $Insert(T, i)$ inserts the interval i to the dynamic set T.
2. $Delete(T, i)$ deletes the interval i from T. If T did not contain i, then this operation makes no change to T.
3. The Min (T) operation on a dynamic set of intervals returns an interval with the smallest value of the *lo* field among all the intervals in the set.
4. The $Max(T)$ operation, returns an interval with the largest value of the *hi* field.
5. The operation $LoSucc(T, i)$ takes an interval i and returns the interval that follows this interval in the sorted order of all the intervals by *lo* field (or returns NIL).
6. The operation $HiSucc(T, i)$, takes an interval i and returns the next interval in the sorted order by the *hi* field (or returns NIL).
7. The operation $IsOverlap(T, q)$, where, i is a given query interval returns 1 if q overlaps with some interval in T and is 0 otherwise (i.e., q does not overlap with any interval of T).

Input: The input will be an interleaved sequence of operations defined above encoded in the following way.

1. Intervals may be input prefixed by $+$ (for insertion) or $-$ for deletion. Interval coordinates are real numbers of the from l h (with whitespace in between). It may be assumed that $l < h$. However, l, h may be positive or negative.
2. The min operator is specified as min with whitespace before and after. Similarly, the max operator is specified as max .
3. The operator *LoSucc* is specified as lsucc (with whitespace) and similarly *HiSucc* is specified as hsucc.

4. The operator *IsOverlap*(i) is specified as overlap (with whitespace) followed by the query interval l h.

The input is a single line. To process the input, think of the input as a sequence of commands, starting with + or − or *min* or *max* or *lsucc* or *hsucc* or *overlap*. Each command, depending on its definition, will take some argument(s) as defined above. The + and − commands do not yield any output. The other operators give outputs as defined. The operators *min*, *max*, *lsucc* or *hsucc* return an interval with the syntax $[l$ $h]$ (whitespace in between). The operator *overlap* returns 0 or 1 as per its definition.

Example. Consider the input.

```
+ 1 5 + 2 4 +3 8 + 11 13 min hsucc 2 4  + 12 20  max overlap 9 10
```

Explanation. First insert $[1, 5]$, then insert $[2, 4]$, next insert $[3, 8]$, then insert $[11, 13]$. Now find the interval with the minimum lo field, which is $[1, 5]$; next find the *HiSucc* of $[2, 4]$ which is $[1, 5]$. Next insert $[12, 20]$, then solve max, which is $[11, 13]$; and finally, check for overlap with the interval $[9, 10]$, whose answer is 0. The output in a single line is

```
[1 5] [1 5] [11 13] 0
```