Lab 9: Red-Black Tree

In this lab, we wil implement red-black tree ADT. In particular, we will implement *insert* function.

1. Input

Obtain a list of numbers from the given input file, and execute an insertion operation for each number in order. At each iteration of insertion, print the red-black tree by using inorder traversal. An example input file is shown below.

Input.txt

|  |
| --- |
| 7 5 3 10 23 4 20 21 22 23 24 25 |

2. RB tree ADT

(1) Data Specification for the objects

struct RBNode;

typedef struct RBNode \*RBTree;

typedef struct RBNode{

ElementType Element;

int red; /\* red=1 when the node is red \*/

RBTree left;

RBTree right;

RBTree parent;

}RBNode;

(2) Function specification

* RBTree Insert( ElementType X, RBTree T )
* printLevelByLevel(RBTree T)

3. Program description

* name : p9.c
* input : a list of operations in a file (an input file name is given as a command line argument. See the example in “1. input” on the first page)
* output : the corresponding result in the standard output

Submit to the course website ([https://portal.hanyang.ac.kr](https://portal.hanyang.ac.kr/)) your source code and a written report. Your report should include the description of your own implementation.