Data Structures B
FAST-NU, Lahore, Springs 2016

Homework 7

2-3 Tree Implementation

Due: Wednesday March 30 11:55PM Marked out of 50 points.

This is a straight forward assignment. You will be implementing a template 2-3 Tree, reading data into it from a file called students.txt, allowing the user to add and remove students from the tree; and providing different kinds of view and update queries.

1. The class Student and the input file ‘students.txt’

Here’s a skeleton for the class Student (you can add to it, if needed).

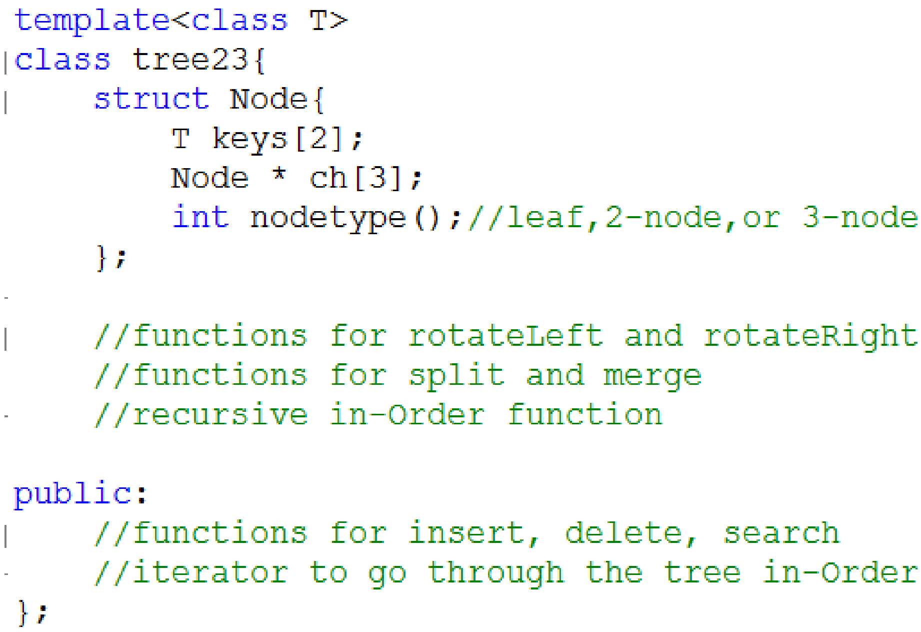
```
class Student{
    string fname, sname;
    string rno;
    float cgpa;
    int batch;
    int wcount;
public:
    Student();
    Student(const string&,const string&,const string&,float,int,int);
    friend operator << (ostream&,Student&);
    bool operator <= (const Student &);//compare roll-num
    bool operator == (const Student &);//compare roll-num
};
```

“Students.txt” contains student records in a line-by-line format, with one student record per line. Each line looks like the following:

first_name,second_name,roll_number,cgpa,batch,warning_count
e.g. Salman, Ahmed, L154404, 3.91, 2015,0

2. The class tree23

Here’s the skeleton for the class tree23 (add to it as you see fit).



```
template<class T>
class tree23{
    struct Node{
        T keys[2];
        Node * ch[3];
        int nodetype();//leaf,2-node,or 3-node
    };

    //functions for rotateLeft and rotateRight
    //functions for split and merge
    //recursive in-Order function

public:
    //functions for insert, delete, search
    //iterator to go through the tree in-Order
};
```

3. Functions in the Menu

Print a menu on the console, with the following options. These will be global functions using a tree23 object created in the main.

i. Read Student Data from file.
input: file name
ii. Save Student Data to a file
iii. **input:** file name
iv. Print all students in order of roll numbers (line by line)
v. Print students with top 10 GPAs (you can use a max-heap to do this).
vi. Print all students ordered by their batch (use an appropriate extra data structure to do this, assume that batches are only from 2010 to 2015)
vii. Print all students with warnings (first, all the students with one warning, then the students with two warnings)
viii. Remove a student with given roll-number
input: roll number
ix. Add a student with given roll-number
input: roll number
x. Remove all students of a batch
input: batch
xi. Change the roll-number of a student to new roll-number
input: old roll number, new roll number

That’s it ☺