CS2303 Systems Programming Concepts A14

Lab 4 {August 25, 2014} 5 points

Binary Trees in C or C++

Program 4 involves building a database of items in RoboMall using a binary tree. Lab 4 is intended as a simple step towards the fourth programming assignment. This program can be completed using C or C++ because we may not cover enough C++ in lecture prior to the lab. However, since Program 4 MUST be done in C++ you are encouraged (but not required) to use C++ for this lab. The form of the input to Lab 4 is similar to Lab 3 with one command line argument:

```
./lab4  students
```

and an example is:

```
./lab4   45
```

The Lab 4 program reads in students lines of input from a script file where each line of input takes the form:

```
student_id   g1   g2   g3
```

where student_id is a 9-digit integer student id and g1, g2 and g3 are student test scores between 0 and 100 on three assignments. The test scores are to be read in as floating point numbers.

The Lab 4 assignment is to create at least two functions to operate on a binary tree: insert and print_tree.

```
insert (node, binary_tree)
```

The insert function builds and maintains a binary tree in student_id order. insert takes one student's information contained in a node and inserts that node for that student in the correct place in the binary tree. Each node should contain the student_id, the three test scores and the average test score.

Note – For this lab you need to add some validity checks of the input data. Assume that student_id is unique. Thus, there should only be one student with a given student_id.

print_tree (binary-tree)

The print_tree function traverses the binary tree and prints out all the nodes in the binary tree in student_id order. Print out the information one line per student and include a header line to identify the columns of the output.

Lab 4 main program

The Lab 4 main program reads students input source lines. For each input line, it creates a binary tree node, and inserts this node in the tree. Once the tree is completely built, the program prints out **all** of an individual student's information in student_id order from lowest to highest number.

Lab 4 Assignment

0. Prior to coming to the lab prepare a preliminary solution to the program above.
1. Create a make file.
2. Test the program under your own test data input from the terminal or a file.
3. Run the program on the provided test data file 'lab4.dat' redirecting the output to binarytree.txt.
4. Create a README file that contains any useful information to assist in the grading of your lab program.
5. Create a tarred file that contains all the source and header files, the make file and the README file and your output file.
6. Use the Linux version of 'turnin' to turn-in the tarred file. [The deadline for all lab turn-ins is 24 hours after the beginning of your assigned lab.]