Lab 6 – due Wednesday, 10/25 at 11pm

Red Black trees

1. (15 pts) Create the Red Black tree (left leaning) for D E S I G N C O D E T E S T. Show all steps.
2. (60 pts) Create or find (on the internet/book) an existing Red Black Tree data structure. It must be the implementation of a 2-3 tree as a left leaning red black tree. and it must be generic for both the key and value. It must include the methods: get(), put(), keys(), RotateLeft(), RotateRight(), flipColors(), isRed(), and getColor(). If it has any extra methods, you must remove them. The file lab6in.txt has two types of lines:
    a) type variable_name; (only 2 types: float or int – set initial value to 0.0 or 0).
    b) name = value;

    Write a driver (main) that reads each line and either put the new variable in the red black tree or change the value of the variable that is already in the tree. Print out each key (the variable) and value (the value of the variable and the data type of the variable) in the tree and color of the node. NOTE: You do not have to put error handling in your code. In other words, assume the input file has only the proper lines as above. You must create a separate class for the variable name, type and value. This class will be the value in your RedBlack Tree.

    Here is my output. I expect you to have the same colors and order:

```
area            float 32.8    Red
circumference   float 0.0     Black
dimension       int   0       Red
max             int   1000    Black
min             int   1       Black
rbtree_size     int   10000   Black
rfl             float 5.5     Red
sfl             float 6.5     Black
temperature     float 98.6    Black
tfl             float 0.0     Black
var             int   12      Black
xvar            int   5       Black
yvar            int   7       Black
zvar            int   0       Black
```

3. (25 pts) Using the same RedBlackTree data structure from problem 2, write another driver/client that will do the same thing as lab 6 question 1. You must get the same answer as your picture in question 1 above (most important – problem 1 and output show same color of node). Need key, value, and color.

I can't show you the output because it gives away the answer to problem 1.

Submission:

Rubric for problem 1:

- 10 pts – created with Red Black tree with all steps
- 5 pts – have the correct final Red Black tree

Rubric for problem 2:
- 5 points – comments explaining what every task or set of computations is doing – this is the proof that you understand the code you picked up or wrote.
- 10 points – program runs
- 5 points – program reads from the file lab6in.txt
- 10 points – The output shows the right red or black (to match my output) for each variable
- 5 points – output shows the correct inorder traversal given the required input.
- 5 points – red black tree data structure has the required methods listed: get(), put(), and keys(), RotateLeft(), RotateRight(), and flipColors()
- 5 points – the output shows the correct data type and value for each variable
- 5 points – Comment that shows where the Red Black Tree came from
- 10 points – driver (main) you wrote from scratch
- You will get a 0 if you:
  a) Did not submit the source code
  b) Did not submit the screenshot
  c) Did not make your red black tree generic

Rubric for problem 3:
- 10 points – program runs correctly using same input from question 1
- 10 points – The output shows the right red or black (to match my output) for each variable
- 5 points – output shows the correct inorder traversal given the required input.
- You will get a 0 if you:
  a) Did not submit the source code
  b) Did not submit the screenshot
  c) Did not use the same red black tree data structure from problem 1