Lab 06 Specification – Red Black Trees Implementation  
Due Friday, 13th April 2018 1PM  
Total - 50 points.

This is a team based lab, so you are allowed to work together with your team mates.  
Please refer to the syllabus for information about late submission policy.  
This lab is related to the textbook Chapter 3 pg: 432 - 447 reading and the lecture slides 18, 19, and 20 review.

Lab Goals

- Implement a Red Black Tree Insertion procedure.  
- Implement a Red Black Tree Deletion procedure.  
- Integrate your Red Black Tree implementation with the JGraphT visualization library.

Assignment Details

A redblack tree is a kind of self-balancing binary search tree in computer science. Each node of the binary tree has an extra bit, and that bit is often interpreted as the color (red or black) of the node. These color bits are used to ensure the tree remains approximately balanced during insertions and deletions. (Definition taken from wiki)

Part One: Red Black Tree Insertion Procedure

Develop a method called RBTreeInsert that performs node insertion into a Red Black tree structure. Initially the RB tree will be empty. The tree is built as the nodes are inserted by the user. The implementation should include the logic outlined below:

1. Your program would need to prompt the user to input "if the node value is either a character or an integer value".
2. It is required that you implement a tree data structure, which allows one to easily represent a Binary Search Tree.
3. In your implementation layer, you would need to include all the different red black tree insertion rules discussed in class.
4. The critical component of the insertion procedure is the node balancing. Your implementation needs to incorporate the RB insertion rules to perform the balancing act during the process of every node insertion.

Part Two: Red Black Tree Deletion Procedure

Develop a method called RBTreeDelete that performs node deletion in a Red Black tree structure. Initially the RB tree would be non empty. The tree is adjusted as the nodes are deleted by the user. The implementation should include the logic outlined below:
1. In your implementation layer, you would need to include all the different red black tree deletion rules discussed in class. The deletion rules include base case, terminal cases, and non terminal cases.

2. The critical component of the deletion procedure is the node balancing. Your implementation needs to incorporate the RB deletion rules to perform the balancing act during the node deletion.

**Part Three: RB Tree Visualization**

Develop a method called RBTreeVisualizer that integrates the JGraphT library into your lab code for displaying the red black tree structure in a Java Applet.

1. An example of JGraphT implementation of the Red Black tree is provided to you in the source code available for download through the course webpage.

2. In order to compile the code, you would need to run the following command from the JGraphT folder:
   ```
   javac -d ../bin/ -cp ..:/Users/amohan/Desktop/JGraphT/jar/*
   /Users/amohan/Desktop/JGraphT/src/com/web/*.java
   ```

3. In order to run the code, you would need to run the following command from the JGraphT bin folder:
   ```
   java -cp ..:/Users/amohan/Desktop/JGraphT/jar/* com.web.RBTreeVisualizer
   ```

4. An important part of this implementing this section, is you need to adjust the X and Y coordinates automatically based on your tree structure as the tree grows and shrinks. The sample code has hard coded implementation of the X and Y coordinates for the different nodes in the tree.

5. The color of the nodes can be modified through the code. The sample code has hard coded implementation of the color specification for the different nodes in the tree. Your implementation needs to modify the node color on the fly, based on the Red Black tree structure.

**Extra Credit**

An extra credit of 5 bonus points will be given to those who add the following logic to your RBTreeVisualizer component of the Lab:

1. In the current implementation, each node is represented using a rectangle box. Change this in the code to represent the node using a circle instead.

2. In the current implementation, each edge is represented using the (source, destination) label connected to it. Modify the code to disable the label, and have the edges displayed without any label.

   In order to do add these two functionality to your code, you may need to do some research by reading the documentation and JavaDocs from:
   ```
   http://jgrapht.org/
   ```

The 5 bonus points will be directly added to your total points for the Lab, and may have a positive impact to your final grade.
Submission Details

1. You are required to submit this lab by sending an email with a zipped version of your cmpsc250-lab06-YOURFirstInitialLastName folder.

2. Subject of your email should say "CMPSC250: Team X Lab 06 Submission". Here X needs to be replaced with your group number. Refer the group sheet in the course webpage.

3. One email should be sent for your teams lab submission.

4. Send the email to amohan@allegheny.edu // CC the email to all your team members.

5. You should add the following statement in the body of your email
   By doing this submission, I understand that I and my team members are subject to the Honor Code policy.
   Lab submitted by: X1, X2, and X3 (Here X1, X2, and X3 are the name of your team members)

6. Shifting team members is not ideal and not allowed, unless there is an extreme situation which is discussed with the instructor prior to your submission.

7. Provide README file inside your code, that will serve as your documentation and it should give details on how to run your program, give a short description of your algorithms and their worst case complexity and any additional sources you have used. Only one submission from one of the group members is needed, but make sure all of the group members’ names are on the submitted documents.

Points Distribution

The breakdown for this assignments points is :

- Part one: 15 points
- Part two: 15 points
- Part three: 15 points
- Documentation (README file) : 5 points

Note

- Although this is not due next week, I expect you to be in the lab next week to continue working on your lab and to ask any questions or clarification needs you may have.