CS 5353/7353 – Analysis of Algorithms
Project – 100 points
Due at beginning of class on Th/04/21/22

Submit a .zip file containing source code and output for Question 1 and a document file for your comment/responses to Question 2.

(a) Implement the abstract data type red-black tree using a programming language of your choice.

Your implementation should include RB-INSERT(T,z), RB-DELETE(T,z), and RB-DISPLAY(T) and any supporting routines. RB-DISPLAY(T) should display the structure of the tree in some reasonable fashion (a simple in-order transversal does not reveal the structure) along, of course, with the key and the color of each node.)

(b) Display (RB-DISPLAY) the red-black tree resulting after successively inserting the keys 41, 38, 31, 12, 19, 8 into an initially empty red-black tree; subsequently, delete 12, insert 32, and delete 41, displaying the structure after each step.

(c) Subsequently, and again starting with an initially empty tree, display the red-black tree resulting after successively inserting the keys

```
   834,807,512,882,127,675,75,216,822,249,114,689,625,974,221,92,374,
123,838,930,654,806,234,381
```

and, then deleting 127 and 221.

2.

(a) Suppose that a node x is inserted into a red-black tree with RB-INSERT and, then, is immediately deleted with RB-DELETE. Illustrate with an example that the resulting red-black tree is not necessarily the same as the initial tree.

(b) The number of black nodes on any simple path from, but not including, a node x down to a leaf is called the black height of the node. Show that one can maintain the black heights of nodes as fields in the nodes in a red black tree during an RB-INSERT procedure without adversely affecting the asymptotic performance of the procedure.