CSC 226 SPRING 2019
ALGORITHMS AND DATA STRUCTURES II
ASSIGNMENT 1 - PROGRAM
UNIVERSITY OF VICTORIA

1 Programming Assignment

The assignment is to design and implement a program that computes the percentage of red nodes in a given red-black tree (not counting external nodes). You will test your program by running at least 100 trials of the experiment of inserting n random keys into an initially empty tree, for $n = 10^4, 10^5$, and 10^6.

Input: A random sequence of n distinct integers.

Output: The percentage of red nodes after inserting all n integers into a red-black tree in the given order.

You will accomplish this by using the textbook's RedBlackBST java class as a template. You will only need the code for the Node class, and the put(), isRed(), flipColors(), rotateLeft(), and rotateRight() methods (the delete() methods are not needed.)

Your task is to add to this a method called percentRed() which returns the percentage of red nodes in the red-black tree as an integer between 0 and 100. How this is accomplished is up to you. You may alter the other methods as needed to accomplish this goal. Keep in mind, the more efficient your percentRed() code is at doing this the better.

You will also need to add a main() method, which should do one of two things:

1. If one exists, read the contents of a text file provided on the command line, for example

   ```
   C:\> java RedBlackBST test_file.txt
   ```

   for correctness testing by the marker.

2. Else, generate your own sequences to accomplish your experiments for the different values of n given above.

2 Test Datasets

A set of input files containing test data are available in the 'Assignments' folder under the 'Resources' tab on ConneX, sorted by their size. You should ensure that your implementation gives the correct answer on these test files before submitting.

My outputs for the test files:

1.
   ```
   Reading input values from test10.txt
   Percent of Red Nodes: 20
   ```

2. Reading input values from test100.txt
   Percent of Red Nodes: 3

3. Reading input values from test1000.txt
   Percent of Red Nodes: 1

3 Evaluation Criteria

The programming assignment will be marked out of 25, based on a combination of automated testing and human inspection. I would like you to hypothesize on the expected percentage of red nodes in any red-black tree and report your results in a written document to be handed in with the program. That 5 marks is included in this program mark.

Score	Description
0 – 5	Submission does not compile.
5 – 10	Compiles but incorrectly reports percentages.
10 – 15	Correctly calculates the percent red but not efficient as possible. That is, `percentRed()` runs in $O(n)$ time or worse.
15 – 20	Correctly and efficiently calculates the percentage of red nodes. That is, `percentRed()` runs in $O(1)$ time.
+5	Hypothesis and results included

To be properly tested, every submission must compile correctly as submitted. **If your submission does not compile for any reason (even trivial mistakes like typos), it will receive at most 5 out of 25**. The best way to make sure your submission is correct is to download it from ConneX after submitting and test it. You are not permitted to revise your submission after the due date, and late submissions will not be accepted, so you should ensure that you have submitted the correct version of your code before the due date. ConneX will allow you to change your submission before the due date if you notice a mistake. After submitting your assignment, ConneX will automatically send you a confirmation email. If you do not receive such an email, your submission was not received. If you have problems with the submission process, send an email to the instructor before the due date.