TCSS543A – Winter 2017
Homework 3 (30 points)
Programming project

This is a group assignment. Each group may consist of 1 or 2 students (*very* exceptionally 3 students, on a case-by-case basis, and with enough justification). In other words, you can choose to work by yourself, or with only one colleague. *Each student must submit their own copy of this homework to Canvas*.

Each group is also expected to give a short presentation and demo of your submission during the last class before the final exam. That presentation must cover all parts of this assignment that you actually executed, and will be graded separately.

Details: In this assignment, you will implement three different algorithmic techniques in **Java**.

Your homework should be electronically submitted via Canvas by midnight on the due date. You are expected to submit the following files:

- Report. The submitted report MUST be typeset using any common software and submitted as a PDF. We recommend using LaTeX to prepare your solution (you could use any LaTeX tools such as ShareLaTeX, TexShop, TeXnicCenter, TeXWorks, etc).
- Java source code. The main class must be called `TCSS543.java` and your program must run a standalone program that prints to the standard output, to be invoked from the command line as:

  ```
  java TCSS543 > output.txt
  ```

- The output file(s) obtained as the result of the above command, containing the data you document in your report.

Remember to cite *all* sources you use other than the text, course material or your notes.

This project involves implementing Brélaz's `Dsatur` heuristic algorithm to obtain an approximate solution to the vertex coloring problem on graphs, which is known to be NP-hard. Reference: Daniel Brélaz, "New Methods to Color the Vertices of a Graph," Communications of the ACM, vol. 22, no. 4, pp. 251-256, 1979.

1. (10 points) In the worst case, Brélaz's `Dsatur` algorithm has $O(|V|^2 \cdot |E|)$ running time using a naïve implementation. Describe how to improve this by using a careful choice of data structures and supporting algorithms.

2. (10 points) Implement Brélaz's `Dsatur` algorithm. What is the running time complexity using your choice of data structures?

3. (10 points) Generate random graphs using the strategy described in Section 3.1 of Brélaz's paper, with $|V| = 10, 20, 30, \ldots$ vertices (try to go as far as possible). Apply Brélaz's `Dsatur` algorithm to color them and measure the running time (for each value of $|V|$, measure the time needed to color 100 random graphs of that size at once, then divide the time by 100). Plot your results and compare them against your asymptotic worst-case analysis above. What can you say about the observed (i.e. average) running time?

4. (5 bonus points) Plot the estimated (i.e. observed on average) minimum number of colors needed to color the vertices of a random graph as a function of $|V|$. What can you say about the dependence between $|V|$ and that number?