SYSC 2100 Algorithms and Data Structures
Winter 2019
Assignment 4: Queues and Sorting
Due: March 22nd, 2019

All the methods asked below are to be included in a class named ***Assignment4***. **Assignment4.java** is the only file to be submitted.

Name your methods strictly as specified (case sensitive). When marking, the TAs will use a class (***MarkingAssignment4***) that inherits yours (***Assignment4***). The TAs' class calls the methods by the names that are specified here. Therefore it is of utmost importance that you abide by these names (case sensitive). Any discrepancy in the name will cause issues.

Make your methods fail-safe, i.e handle exceptions (try/catch) or use if/else statements for error testing/avoidance wherever appropriate.

1. Write recursive versions of `selectionSort` and `bubbleSort`. Call them `recursiveSelectionSort` and `recursiveBubbleSort` respectively.

Example method specification:

```
public static <T extends Comparable<? super T>>
                            void recursiveSelectionSort (T[] theArray, int n)


public static <T extends Comparable<? super T>>
                            void recursiveBubbleSort (T[] theArray, int n)
```

2. Consider the language

   L={*w*$*w'*: *w* is a possibly empty string of characters other than $,

   $w' = reverse(w)\}$

   For example "abc$cba" is a string of that language. "abc$cbd" is not. Write a Java recognition method for this language that uses both a queue and a stack. Thus, as you traverse the input string, you insert each character of *w* into a queue and each character of *w'* into a stack. Assume that each input string contains exactly one $. Use JCF's implementations of the queue and the stack. Name your method `isInLanguage`.

   Example method specification: `public static boolean isInLanguage (String str)`

3. When you enter characters at a keyboard, the system must retain them in the order in which you typed them. It could use a queue for this purpose. Once the characters are in a queue, the system can process them as necessary. For example, if you had typed an integer –without any mistakes, but possibly preceded or followed by blanks –the queue would contain digits and

possibly blanks. If the digits are 2, 4, and 7, entered as " 2 4   7" (notice the spaces all over the place), the system should convert them into a decimal value 247.
Implement a Java method `convertToNumber` that uses a queue to convert a sequence of character digits into an integer. You can assume that only spaces and single digits are entered by the user.

Example method specification: **`public static int`** `convertToNumber` **`(String str)`**

Submission Requirements: Submit your assignment, **only** the source file **Assignment4.java,** using **cuLearn**. Your program should compile and run as is in the default lab environment, and the code should be well documented. Submit all the files individually **without using any archive or compression**.

Marks will be based on:

- Completeness of your submission
- Correct solution to the problem
- Following good coding style
- Sufficient and high-quality in-line comments
- Adhering to the submission requirements (in particular the naming convention and the submission of uncompressed source files only)

The due date is based on the time of the **cuLearn** server and will be strictly enforced. If you are concerned about missing the deadline, here is a tip: multiple submissions are allowed. So you can always submit a (partial) solution early, and resubmit an improved solution later. This way, you will reduce the risk of running late, for whatever reason (slow computers/networks, unsynchronized clocks, failure of the Internet connection at home, etc.).

In **cuLearn**, you can manage the submission until the deadline, taking it back, deleting/adding files, etc, and resubmitting it. The system also provides online feedback whether you submitted something for an assignment. It may take a while to learn the submission process, so I would encourage you to experiment with it early and contact the TA(s) in case you have problems, as only assignments properly and timely submitted using **cuLearn** will be marked and will earn you assignment credits.