Assignment 1 – (CSCI4731) Software Design and Patterns – Spring 2022

This assignment contains a single task with two parts.

- The first part is to design a class diagram solution to the given problem. To generate the UML class diagram, you are allowed to use any tool convenient to you.
- The second part is to implement the given design structure in Java language using Object- Oriented Programming techniques. Make sure, that your code is neat, clean and thoroughly commented. Try to use meaningful variable names. This assignment requires you to use **the Strategy** and **the Observer** patterns.

  All submissions should be uploaded to Blackboard as a single zip file.

  Deadline: April 12th, 21:00

The Task - Sentence Generator

Description

Sentence Generator is a program that generates sentences using English words. All sentence generators have an internal vocabulary, which is initially empty and can be updated by adding some words. All sentence generators are able to generate sentences according to some rules. You have to implement following three types of sentence generators:

- **Random Sentence Generator - RSG**

  When a word is added to RSG, before storing it in internal vocabulary, the word is converted to lower case. RSG generates new sentences by randomly picking random amount of words from its internal vocabulary and concatenating them using single space between the words.

- **Sorted Sentence Generator - SSG**

  Before adding a word into its internal vocabulary, SSG also converts the word to lower case. Like RSG, SSG picks up the words randomly. The only difference is that it sorts these words before the concatenation.

- **Ordered Sentence Generator - OSG**

  OSG is different from both RSG and SSG. A word, before adding into internal vocabulary, will be converted to upper case and reversed. OSG concatenates all of the words in the same order they have been added to the vocabulary.

  A single class Vocabulary should be responsible for getting user input from console and notifying all sentence generators about a new word. Whenever there is a new word, all sentence generators should add this word to their internal vocabularies and generate a sentence.

Task 1 - 40 points

Generate UML class diagram for this problem. Apply the Strategy and the Observer patterns. **Upload only final image of the UML diagram. Accepted formats: BMP, JPG or PNG**.

Task 2 - 60 points

Implement the solution in Java language using the IDE of your preference (both Eclipse and IntelliJ are welcome). All interactions with the program should be through console. Before uploading, export your project (File – > Export ...) into a zip file and don't just copy the source files from your solution. **Note: all sources codes will be checked for plagiarism. Do not try to cheat!**