**CS264 - Software Design**

Assignment 4

| | |
|---|---|
| Assignment Release Date: | 21-11-2022 |
| Submission Due Date: | 02-12-2022 |
| Feedback Due Date (estimated): | 17-12-2022 (for assignments that make Due Date) |
| Support Laboratories | Labs 08-09 (Two Weeks) |
| Total Mark: | 10% |

*This Assignment is worth 10% of the Software Design CA Component.*

This is an open-book, graded assignment. You may use online resources for reference purposes only to help with the assignment. Please cite all references as comments in your submissions. You cannot directly reuse C# **solution code** from online sources. **You must not engage with another student, in person or electronically (phone, social media, etc.) to secure assistance with this assignment. If you do so you will receive an automatic fail (0%).** We will perform similarity checks on submitted assignments to check for collaborative efforts. A reasonable attempt at this assignment will gain you 10% of your continual assignment marks. It is possible to gain extra credit (up to a maximum of 5%) for this assignment.

**Assignment 04 - Software Design and Design Pattern Implementation with C#**

You are required to re-design and re-develop the console application that you developed in Assignment 03. The existing user interface should be retained, i.e. providing simple keyboard functionality to (i) generate different random shapes that are added to the canvas, (ii) display the current canvas to the console, (iii) save the canvas to a file, (iv) implement Undo-Redo functionality in your interactive session.

**Note that you do not need to implement a graphical user interface for this assignment, i.e. you are not building a graphics app. This assignment is about designing and developing the classes and methods associated with a typical drawing app data model and implementing a Software Design Pattern.**

**Assignment 04 - Requirements**

You are required to implement the following core functionality for this assignment:

1. You must re-implement the Undo-Redo functionality for shape commands using Command Software Design Pattern for this assignment. You must clearly describe in your application comments the pattern you are using and how it is implemented. Failure to specify the pattern design details using appropriate comments, or not using a design pattern will reduce marks allocated.

2. You must include a class diagram (PDF format) for your application. This should be derived from the codebase using, for example, the automated PUML extensions for VS Code. We will use these diagrams to compare the class structure to the software design pattern chosen to implement Undo-Redo.

3. You must include a detailed description (500-1000 words) explaining, comparing and contrasting your core Undo-Redo designs for Assignments 03 and 04. In particular, you should reflect, and comment, on which design pattern worked better for your solution, and why this is the case. You should provide some notes on how you believe your different implementations would scale, for example, if you were to add more commands, or the canvas gets larger. Please note that this reflection **must** relate to **your** solutions, and should not be a general re-presentation of theoretical observations on the software design patterns. If you provide "pros and cons" for the different designs, then they must refer to your solutions (i.e your **context**), and not to the patterns in general. Finally, you should provide short notes on personal challenges associated with implementing the different design patterns.

**Assignment 04 - Extra CA Credit**

There is no extra credit option available for this assignment.

**IMPORTANT SUBMISSION DETAILS**

**Please indicate the Operating System (Linux/Windows/MacOS/Online) and IDE (e.g. VS Code) version used for testing (as a comment in your submitted code).**

All work must be submitted via Moodle (see "Assignments" section for submission). Work submitted via other means will not be accepted unless you have prior arrangements with the Head Demonstrator (Mark McCormack). All work MUST be submitted by the due-date deadline. Late submissions will not be accepted.

*The assignment submission is a zip file named "**assignment-04-xxxxxxxxx.zip**" (where "xxxxxxxxx" is your student id) containing solution files, e.g. named "**svgundoredo.cs**", "**Program.cs**", etc. together with any other resources used in the assignment solution. Please ensure that all external files use relative directory referencing, rather than hard-coding the files' location.*