CS-224 Object Oriented Programming and Design Methodologies

Assignment 04

October 5, 2018

1 Guidelines

You need to submit this assignment on 15th of October at 630 pm as the next assignment will be given on the same day. Some important guidelines about the assignment are as following:

- You need to do all the assignments alone
- You will submit your assignment to git-hub
- You need to follow the best programming practices
- Submit assignment on time; late submissions will not be accepted.
- Some assignments will require you to submit multiple files. Always Zip and send them.
- It is better to submit incomplete assignment than none at all.
- It is better to submit the work that you have done yourself than what you have plagiarized.
- It is strongly advised that you start working on the assignment the day you get it. Assignments WILL take time.
- Every assignment you submit should be a single zipped file containing all the other files. Suppose your name is John Doe and your id is 0022 so the name of the submitted file should be JohnDoe0022.zip
- DO NOT send your assignment to your instructor, if you do I will just mark your assignment as ZERO for not following clear instructions.
- You can be called in for Viva for any assignment that you submit

2 Task

For this assignment you will be creating a drawing program of your own. You are provided with a folder `Artistik` that contains the sample code. You will need to setup the project properties to use `SDL2.0` to make this code work on your machines. How you can set up the project to use `SDL2.0` can be learned by following the lazyfoo tutorials present at this link. Once everything is working, you will be able to draw a red rectangle by left clicking the mouse and dragging. The code is however incomplete as every time you draw a rectangle, the old rectangle is lost.:

In order to fix this issue, you will need to store every rectangle in a stack. As you keep on populating the stack, the number of rectangles will increase on screen. You are also given two structures for Point and Color as well. The tasks you need to accomplish are:

- You will need to declare a **Shape** base class
- Two classes **Rect** and **Line** will inherit from the Shape class. You will need to study SDL's documentation to understand how a line is drawn.
- You will create a **Stack** of your own
- Each node of the stack will only be able to store a pointer of type **Shape**. What this means is that whenever you will create a child object (either Rect or Line) its address will be passed to this base class's pointer. As a result, the stack will be able to hold both Rect and Line objects.
- Every time you will create a line or a rectangle, it will have a random color (use the **Color** object for this reason).
- You will select what to draw by pressing 'r' for Rect and 'l' for line.
- You will undo with right mouse button whatever you draw by popping the stack.
- Whatever shape you pop, will be stored in a separate "undo" stack
- You will redo with the middle mouse button which will pop the value from the "undo" stack and push it in the first stack.

- if you undo and then draw some new shape, it will be added to the stack and subsequently purge the "undo" stack.
- '-' and '+' keys will change the order of the last drawn object within the stack. If you press '-' the shape will go deeper in the stack. If you press '+', the shape will start moving to the opt of the stack.

– The End –