Assignment 7 – Group Project: SDL Graphical Application

Date Assigned: Friday, April 26, 2019
Date Due: Monday, May 6, 2019 – **There is no grace period for this assignment**
Code Points: 40 points
Presentation: 10 points

Goals for this assignment

1. Implement the basics of 2D graphics programming
2. Work efficiently and effectively in small groups
3. Reinforce all of the C++ concepts
4. Use a real API (SDL) in program development

For this final assignment, you will be placed into groups to create a graphical application using SDL. On Monday, May 6, each group will present their application, talking about what each person did and answering any questions from the class.

Application Specifics

You may create any application of your choosing, but it must:

1. use sprites in a significant way
2. be large enough for each person to write a decent amount of object-oriented code
3. have a significant amount of animation
4. allow the user to interact (keyboard and/or mouse) with the application as time passes
5. use dynamic memory allocation in a meaningful way
6. use proper object-oriented design principals including **inheritance** (other than the sprite inheritance) and **polymorphism**

The easiest application to write is some kind of game. Possible game ideas include:

1. Snake (http://www.snakeonline.net/)
2. HangMouse (https://www.spellingcity.com/hangmouse-kids-hangman-online.html?listId=6111855)
3. Asteroids (http://www.mspacman4u.com/asteroids/)
4. Maze (http://www.gamesolo.com/flash-game/maze.html)
5. Space Invaders (http://www.pacxon4u.com/space-invaders/)
6. Brick BreakerI (https://www.coolmath-games.com/0-bricksbreaking)
7. Brick BreakerII (https://play.google.com/store/apps/details?id=com.elbylabs.brickbreakerrestructured&hl=en)
8. Dots (https://www.youtube.com/watch?v=oAb1Xvw-9pQ)

Other ideas could be some kind of graphical application (simulation/tool) used in Biology, Chemistry, Physics, …

Groups

You are each assigned to a 2-3 person group. One of the first things that you should do is exchange contact information. Working on a group programming project can be complex, and it is important that you communicate clearly and often with your group members.

By TODAY **Friday, April 26, 2019 at 11:55pm**, each group must submit the following via email:

1. The name of the application
2. A description of the application
3. How the work will be divided. Describe which person will work on what part of the application.

Make sure your application has enough content for the number of individuals in your group. Divide the work by the number of people in your group as equally as possible. If there is not enough content or the idea is unacceptable, I will email you back requesting modifications.

Use the CS250Groups folder on Grace to share and pass around your project code. More details below

Final Group Presentation (May 6 during class time)

Each group must present their application and explain what each individual did. Each group member must attend and be on time. Each group member must stay for the entire class session. Missing any portion of a presentation will be a loss of 10 points.

Please respect your classmates: turn off your phone, log off the computer, no typing, and please ask good questions.

Each person in a group is to talk for a consecutive period of time. The minimum amount of time is 1 minute 30 seconds and the maximum amount of time is 2 minutes, so you will need to practice. The minimum time your presentation can take is 6 minutes 30 seconds and the maximum time is 8 minutes. You will be cut off at exactly 8 minutes whether you are done with your presentation or not.

Each group member must pick one of the following topics to present, and for groups of two, someone must cover two topics. The topics must be covered in the following order:

1. The UML design of the entire SDL Application and any new SDL material learned. Your UML design may have changed from the one you submitted earlier. Use the corrected design.
2. Are you using polymorphism in your solution? If so, explain how you are using polymorphism and why you decided to do so. If you are not using polymorphism, explain why not. In either case, take us through one or two pieces of code that were problematic during the development process. Make sure the code is visible on the slide from anywhere in the classroom.
3. With the UML code on the screen, explain who did what in the project. Discuss what worked well with the group and what did not work so well. What would your group do differently if you had this to do all over again? Finally, demo the project. Allow for a 1 minute demo. The demo time is included in the 9 minute total time frame.

Solution Details

1. Copy over the spriteLab solution from the CS250 public folder on Grace. Rename the solution folder to 07GroupN, where N is your group number.
2. The solution contains two projects:
    a. SDLManager: do not modify!
    b. SpriteLab: rename your project to your application title
3. Add any other projects as you might need. For example, you might need to use Graphics2D, which you can copy over from your Boomshine project.
4. You will need to link your projects to SDL, the SDLManager project, and any other projects you might have. All the linking information can be found in 09-SDLRectangleLab, which was handed out on March 15 (Week 7). You can also look at the properties of the projects in Boomshine to identify all the necessary settings.
5. On Grace, there is a folder named (CS250Groups), which you can use to place your project. I recommend that you always have a current and working project in CS250Groups on Grace. The project should include a README file, which you update at the end with a brief description of what you did. When you start to work on the project, you should copy the project from Grace, work on it, update the README, and then place it back on Grace.

To complete this assignment, you must submit the following:

1. An electronic copy of your program on Grace

a) You need to follow the coding standards. Make sure that you include the hours you worked on the assignment in your header comments.

b) Make sure that your program builds without errors & warnings and runs correctly. If you get any errors or warnings, double check that you typed everything correctly. Be aware that C++ is case-sensitive. You will lose 10% if there are any warnings and 40% if your program does not build successfully.

c) Once you are sure that the program works, it is time to submit your program. You do this by logging on to Grace and placing your complete solution folder in the **CS250-0X Drop** folder.

d) The solution must be in the drop folder 15 minutes before the time class starts on the day the assignment is due. Anything submitted after that will be considered late.

2. A hard copy of your program

a) The hard copy must be placed on the instructor's desk by the time class starts on the day that it is due. Print the driver.cpp, and all the ./.cpp files.

b) **The hard copy must be printed in color, double-sided, and stapled in the upper left corner if your solution contains multiple pages**.

c) Your tab size must be set to 2 and you must not go past column 80 in your output.

Remember, if you have any problems, come to me straight away
with your project on Grace. Good Luck!!!! :)