Assignment 8 – SDL Graphical Application

Date Assigned: Friday, April 27, 2018
Date Due: Monday, May 7, 2018
Points: 40 (10 points on presentation)

For this final assignment, you will be placed into groups to create a graphical application using SDL. On Monday, May 7, each group will present their application, talking about what each person did and answering any questions from the class.

Application Specifics

Your application is to

1. use Sprites/Images in a significant way

2. be large enough for each person to write an adequate 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. maintain a sorted high score or best time of size N (easily set at compile time) that is displayed after the user is done playing the game for that session

7. 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

Section 01 – 11:45 G#1: Cummings, Robasciotti, Meyers, Rodriguez G#2: Miller, Stewart, Rowe G#3: Sindt, Stevens, Wright G#4: Rivera, Dahl, Davis G#5: Deyoung, Lincoln, Delay	Section 02 – 2:15 G#1: Delaney, Howes, Chung, Rasos G#2: Miyasato, Melia, Bunch G#3: Li, Eng, Lemoine

By **Sunday, April 29, 2018 at 5pm**, each group must send me an email as follows:

1. Subject Line: Section # Group #
2. Body
    a. The name of the application
    b. A description of the application
    c. 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.

By **Tuesday, May 1, 2018 at 5pm**, each group must send me a UML diagram detailing all classes and relationships for your solution. If you use any of my classes such as Sprite, just display the class name (no members) in a SimpleClass UML diagram. Then use accordingly in the rest of your design. As an example, look at Direction2D of the UML design from 4.27.18.

On the day the assignment is due, each group will present their application and explain what each individual did. I will give you more details about the final presentation next week. The class will be able to ask questions of each group.

Outcomes for Assignment 8:

1. Implement the basics of 2D game programming
2. Work efficiently and effectively in small groups
3. Create better designs using UML so that group members can interface code easily
4. Reinforce all of the C++ concepts learned in a graphical environment
5. Use a real API (SDL) in program development

To complete this assignment you must:

1. Create a solution based on what your application is and then create projects (you need multiple projects) as necessary to hold related classes of code. You can reuse code from existing projects. All code is to be original (i.e. created by individuals in your group, not copied from any other source).
2. Every piece of code in the solution is to be completely and correctly documented including any code from my sample in class code. You are free to use any example code from class as long as the code is properly documented and the author (Computer Science, Pacific University) is sited.
3. Your code is to be written using Visual Studio 2017 and placed in the CS250 Drop Box by the start of class on the day in which the assignment is due. A stapled hard copy must be placed on the instructor's desk by the start of class on the day the assignment is due. Remember, print (excluding my code) the driver.cpp, and all the .h/.cpp files.
4. Create a solution called GroupN where N is your group number. Classes are to exist in related projects for easy reusability. That is, do not put all classes (.h/.cpp) in the same project or you will lose design points.
5. **THERE IS NO LATE GRACE PERIOD FOR THIS LAST ASSIGNMENT**