

CS250 Assignment 5

SDL Simple Circle

Date assigned: Monday, March 19, 2018

Date due: Friday, April 6, 2018

Points: 30

Goals for this assignment

1. Read and use existing code
2. Write an object-oriented program using multiple classes
3. Implement the Circle class designed in class
4. Practice with basic 2D graphics in SDL beginning with **SDLManagerV1.2**
5. Understand the linking process and setting up dependencies correctly

Create a new solution called **punetid-Graphics** and THREE new projects named **05_Graphics2D**, **05_Circle**, and **SDLManager**. You must link each of these projects together correctly. We will go over this in class.

Make sure you set the project dependencies correctly!!!!

05_Graphics2D must contain **Circle.h**, **Circle.cpp**, and **Graphics2DDriver.cpp**.

05_Circle must contain **CircleTestDriver.cpp** and the specified text data file **circles.txt**.

You must finish implementing the Circle class we design in class (in Circle.cpp).

Your **driver** (CircleTestDriver.cpp) must:

1. Create an instance of the SDLManager called rcSDLManager
2. Create a window of size 640 x 640 at position (300, 300) on the screen with title Simple Graphics
3. Initialize the world correctly. The background is TEAL.
4. Read the data file **circles.txt** which will contain at most 10 circles into an array of Circles.
5. Place each circle read on the screen. You are to have all circles displayed inside an animation loop that continues until the user selects the close box (i.e. the X in the upper right of the window)

Here is how to proceed:

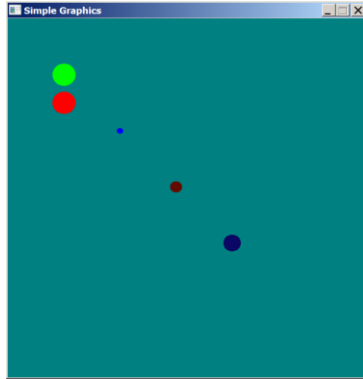
1. Create the SDLManager project in your **NEW** assignment solution.
2. Copy Color .h/.cpp, SDLManager .h/.cpp, and the SDLDriver.cpp into new files in the proper folders of your new solution.
3. Set up the proper paths in VC++ Directories and Linker input files as discussed in class. Then run the simple driver in SDLManager to display a black circle on the screen. If this doesn't work, see me BEFORE writing any additional code. If you've written additional code prior to linking correctly, I will not help you solve your problem.
4. Create the project 05_Graphics2D and add a class Circle with a default constructor only to the interface and the implementation that does nothing in the .cpp. See step 3. Also, 05_Graphics2D depends on the SDLManager, so set this up correctly. Finally, copy the SDLManager driver code into a file Graphics2DDriver.cpp. You will need this driver to compile the Circle code correctly.
5. Copy Circle.h from grace into 05_Graphics2D, implement, and test.
6. Complete the rest of this assignment by adding a project called 05_Circle and writing the code described for CircleTestDriver.cpp.

Notes on Design

You must use the Circle.h provided on Grace that we designed in class. Later projects will rely on the Circle class so be sure to start early and test your code well.

Error Handling

The data file will not contain any errors and will always contain at most 10 circles.



Sample File: The file will end with a blank line

```
100 150 20 255 0 0 255
100 100 20 0 255 0 255
200 200 5 0 0 255 255
300 300 10 100 11 0 255
400 400 15 11 7 100 255
```

The values in the file are:

xposition yposition radius colorR colorG colorB colorA

To complete this assignment you must submit the following:

1. An electronic copy of your program on Grace

- Add a project named **05_Graphics2D, 05_Circle, and SDLManager** to your **NEW** assignment solutions folder. It is vital that you name your solution and your projects correctly!
- Type your program (fully documented/commented) into the project. You need to follow the coding standards from the CS250 Web page. These coding standards have been modified to include additional C++ language features introduced in CS250, so please be sure to read the new coding standards. Make sure that you include the hours you worked on the assignment in your header comments.
- Your program's output must look **exactly** like the sample output above.
- 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.
- 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.
- The solution must be in the drop folder by 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

- The hard copy must be placed on the instructor's desk by the time class starts on the day that it is due. Print in order: CircleTestDriver.cpp , Circle.cpp, and Graphics2DDriver.cpp.
- The hard copy must be printed in color, double-sided, and stapled in the upper left corner if your solution contains multiple pages.**
- 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!!!! :)