

Programming Project #1
CpSc 4160/6160: Data-Driven 2D Game Development
Computer Science Division, Clemson University
Writing Classes for SDL Animation
Brian Malloy, PhD
January 19, 2016

Due Date:

To receive credit for this assignment, your solution must meet the requirements specified in this document, and be submitted, using the web `handin` command, by 8 AM, Wednesday, February 3rd, 2016. If you cannot complete the project by the due date, you can receive 90% of the grade if you submit the assignment within one (1) week after the due date.

Project Specification:

Your assignment is to build a project that implements a simple sprite animation using the Simple DirectMedia Layer gaming framework (SDL). To illustrate the idea, several animations will be shown during lecture. To implement your animation you must use two well-written C++ classes. For example, you might consider writing a manager class that manages the assets and the sprites. You might also consider writing a sprite class that loads and manages the image(s) and moves the image(s) in the animation. You might consider including methods in the sprite class for drawing and updating the image(s). However, the overall design is completely your choice, with the exception that you must include at least two well-written classes that conform to the rules and policies outlined in the slides and videos that I have provided, and described by Meyer's in *Effective C++* (at least the Items that you have been assigned).

You must use the web `handin` command to submit a compressed directory containing your program files, assets, and a README file. Of course your program files must be written in C++ and use SDL. There is no upper bound on the number of classes but please keep the scope of the project reasonable.

Finally, a C++ class will be provided for you that will generate frames so that we can make a video illustrating your animation for all members of the class to view and appreciate.

Projects compressed using `rar` will receive a zero and will not be graded. To compress the directory that contains your project, assume that all items are in a directory labeled 'asg1'. Then you can compress the directory in 1 of 2 ways:

```
tar zcvf asg1.tar.gz asg1
zip -r asg1.zip asg1
```