

# CS 770/870

## Assignment 1: Drawing Shapes

September 3, 2008

**Due: Monday, 9/15 by 23:59:59**

**Last allowed submission: 9/19 23:59:59**

The purpose of this assignment is to give you a fast introduction to the basic features of OpenGL within a simple object-oriented context for representing objects that can be displayed. Using the framework suggested by the demo program, you need to implement a program with the following features:

1. It supports at least 3 classes representing simple shapes. These classes should inherit from the Shape class and implement its behavior.
2. A menu allows the user to select whether subsequently created shapes will be drawn using polygons that have just an outline (boundary only), only an interior (fill only), or both. That is, a single Shape class can be composed of multiple polygons, but all will have the same boundary/fill characteristics.
3. Another menu lets the user choose from a list of available shapes to identify a current shape class. Mouse handling code allows a user to define a rubber band rectangle. That rectangle defines the size and location at which you should create and display a new instance of the currently selected shape.

You are allowed to add functionality to the *Shape* class and/or change the existing implementation of features. In fact, we're expecting you to; *Shape* is pretty minimal.

### Point allocation

- 45 Implementation and display of 3 classes that inherit from Shape that draw 3 distinct shapes using multiple colors and features. You can demonstrate this functionality by hard coding the creation of an initial scene contain multiple instances of each of the 3 shapes.
- 15 Implement the boundary/fill option menu. You can demonstrate this functionality (without the remaining points) by interpreting a selection from this menu to re-draw your initial scene with the newly selected option.
- 15 Current shape selection and rubber band rectangle used to define position at which a new instance of the current shape will be drawn.
- 15 Rubber band rectangle also determines the size of the shape to be drawn

Features that might qualify for the last 10 points include particularly good shapes, deletion of existing shapes using the mouse, changing the location of an existing shape, and many more.

### Note:

Points are earned by correctly implementing features robustly. Points are deducted for bugs, incorrect implementation, poor style, poor design decisions, etc.