UNIVERSITY OF CALIFORNIA, MERCED

BERKELEY · DAVIS · IRVINE · LOS ANGELES · MERCED · RIVERSIDE · SAN DIEGO · SAN FRANCISCO

SANTA BARBARA · SANTA CRUZ

CSE 165/ENGR 140: Introduction to Object-Oriented Programming
Mini Project 2

Due Date: Friday, February 23, 2018

Total Points: 100

1. Download the file GlutAppTemplate.zip from CatCourses and run it on your computer. This is a bare bones GLUT application. There is a sample App class declared in App.h and implemented in App.cpp. You should only modify these two files and possibly add new ones. There should be no need to edit any of the other files included in the project. You may study GlutApp.h to see an example of encapsulation, namely the GLUT functions and data have been bundled into an object.

2. Create a class (struct) named `Rect`, which represents a 2D rectangle. It should store the $x - y$ coordinates of the top-left point of the rectangle as well as its width and height. These should be stored as floating point values. In addition to all the constructors, destructors, getters and setters, your class should have a public method with the following signature:

   ```
   bool Rect::contains(float x, float y)
   ```

   This method should return true if the point specified by the arguments (x, y) is inside the rectangle, and false otherwise.

3. Your app should maintain a list of `Rect` pointers. You can initialize that list with several rectangles in the `App` constructor. The rectangles should be drawn on the screen when the application starts.

4. Each time the mouse is clicked, you should check to see if the mouse pointer is inside one of your rectangles. If a rectangle has been clicked on, that rectangle should be highlighted in some way (drawing a border around it or changing its color).

5. Upload a ZIP archive of your completed project folder on CatCourses and present your code to your TA before the deadline.

Grading policy:

Correct use of the GlutApp template	5 points
Correct implementation of Rect class	45 points
Correct implementation of mouse click handler	45 points
Code style and readability	5 points