Project 3 part c: Out of 50 possible points. Due Nov 27, 2018 11:59pm

BONUS if you submit before Thanksgiving by Tuesday Nov 20, 2018 11:59pm!

Purpose: is to demonstrate understanding of the camera and projection when using raytracing.
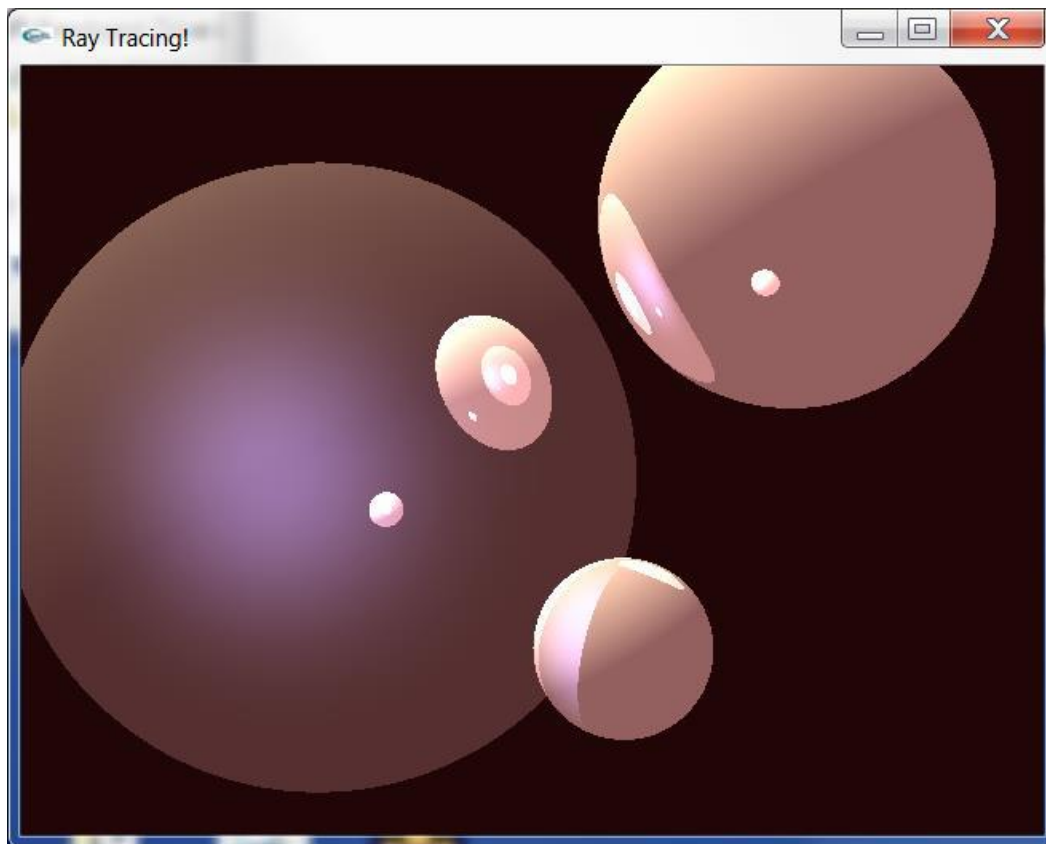
Use code from Proj 3 part b.

Figure 1: result from Project 3

1) Add Reflection. Compute the ray for the reflection using the reflection formula.

2) Make your shade function recursive so you can add multiple levels.

3) Add the implementation to spawn multiple reflection rays- remember that you will need to have a recursive level cap. To being set to level

3, but incorporate a keyboard command '+' and '-' to increase or decrease interactively the level. At various levels your program should look like Figures 3+.

Additional Tasks for 5450:

4) Add Refraction rays using the refraction formula.

Bonus for all: Implement an isInShadow function. In your shade function, implement the calculation of shadow ray, spawn shadow feelers, and appropriately adjust the color components so that shadows can be added to your program. You should implement a keyboard command toggle to turn on and off shadows – "s" should toggle these on and off.

REMEMBER: You should not be using OpenGL at all for transforming objects, defining objects, or displaying objects (other than pixmaps).

You will be graded on elegance of code and the completion of implementation details for elements specified. Elegant code should include programming concepts and formation of classes and functions that you have learned in Data Structures and Algorithms course. Point values are distributed among the categories as follows:

4450:

Element 1 – 20 points

Element 2 – 10 points

Element 3- 20 points

5450:

Element 1 – 20 points

Element 2 – 10 points

Element 3- 10 points

Element 4- 10 points

Start Early! Ask Questions! Don't forget to comment your code!