

Graphics Programming Project 4 Report

Luke Miller
010785453

Problem Statement:

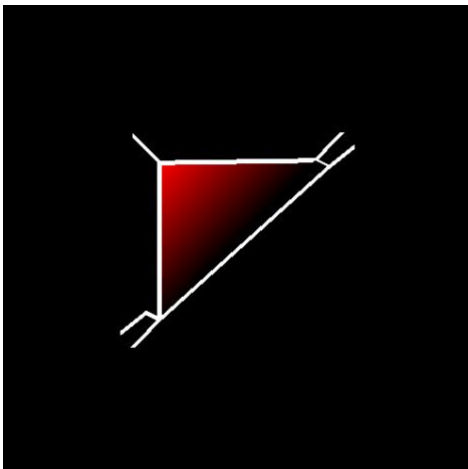
The goal of the programming assignment is to create a formula-based toolbox for geometric operations, interpolation operations and shading operations. This is to give us a first hands on with interpolation and shading. The inputs of the program are the user inputs coefficients for the shading as well as light source and also an input file. There are few places of error handling that keep the user from giving the wrong file name and such but as much to be expected due to the fact that it was not a requirement for this particular project.

Design:

When I designed my program, I first decided to break up the required function bullet points into functions and decide the parameters for them as well as what would be returned for said function. So, there are 5 sections for the program: global class/variables, geometric operations, interpolations operations, shading operations, and utility operations. The main three are as describe in the project instructions, utility is a place for the main as well as a few other helper functions, and global class/variables holds the data structure for the surfaces which was taken from the previous homework.

Implementation:

To get started I looked for widgets that would be useful to the project in the class's source folder. Then I copied the some of the default code from the 'show_poly.cpp' file for the file I/O functions I needed. After that I decided to pretty much get at it with how I implemented the functions with formulas mostly from the slides and a few from internet resources. The hardest part was trying to figure out the correct way of doing the interpolation functions as the instructions weren't very clear and I still do not know if they are implemented correctly or not. I ended up not being able to figure out a good way of splitting the original 4 points into the 16 points I need so I wrote some of the worst code I have ever written, hardcoding the points in my embarrassing splitObject() function. Otherwise, my other functions are really well designed and versatile.



Testing:

After each part was programmed, I checked to make sure each function was outputting probable results. I also tested to make sure my file I/O worked as expected pretty extensively. The interpolation functions are where I ran into most of my problems for it wasn't clear what each required. I'm not quite sure what the output is supposed to look like, but my output of the input file looks like it most likely got interpolated wrong. I email the professor two days ago and tried to attend office hours. The email I got I think was pretty helpful in clarifying my interpolation functions, but I didn't make it into office hours for a few other questions I had because I didn't get

in the queue fast enough before his hours ended with previously queued meetings. I think my shading works well though because there is a nice gradient on my weird object that got outputted, so my program I think mostly works well other than how my object got split up.

Conclusions:

In conclusion, this program was mostly a success. Everything works almost as expected. I am not proud of the hard coded function in my program, but I couldn't figure out a way to automate the work through a loop. Knowing what I know now I would have asked the professor in class about the interpolations functions explanation of what they should do. If it weren't for that the interpolation functions would have taken me less than 3 hours. It ended up costing me around six because the interpolation stuff stumped me pretty hard.