CAP-5701 Computer Graphics

Assignment 2

FIU KF School of Comp. & Info. Sciences

1 Exercises (40 points)

1. (10 points) Write a complete program for displaying a bar chart. Input to the program is to include the data points and the labeling required for the x and y axes. The data points are to be scaled by the program so that the graph is displayed across the full area of a display window.

2. (15 points) Calculate the result of blending the following colors, assuming that source blending factor is equal to source color's alpha value and sum of source and destination blending factors is one (write down the calculation steps).

   (a) (5 points) destination = glColor4f(0.1, 0.2, 0.8, 0.5), source = glColor4f(0.7, 0.3, 0.1, 0.3)

   (b) (10 points) There are four layers: 1, 2, 3, and 4. The i^{th} layer is the destination for $(i+1)^{st}$ layer.

   - Layer 1 = glColor4f(0.3, 0.3, 0.3, 0.1)
   - Layer 2 = glColor4f(0.7, 0.1, 1.0, 0.8)
   - Layer 3 = glColor4f(0.1, 0.3, 0.5, 0.6)
   - Layer 4 = glColor4f(0.9, 0.6, 0.4, 0.4)

3. (15 points) Use Bresenham's algorithm to digitize/rasterize the line with endpoints (5, 24) and (15, 29)

2 Programming Assignment (60 points)

For this programming assignment, you need to complete the program posted on Canvas. The given program provides a graphical UI for the user to

- input convex polygons by moving mouse over the display window
- specify one or multiple fill colors to color the given polygon (one color will result in a solid fill area, while multiple colors are assigned to multiple vertices of the polygon resulting an area filled with linear color gradient)

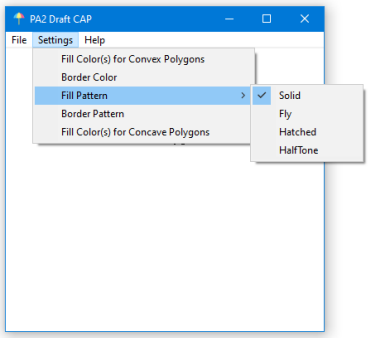

Figure 1: Navigating through the program's main menu.

- pick a fill pattern: solid, fly, hatched, and halftone
- input one or multiple border colors for borders colored either with solid or gradient color.
- input a border line style by providing a four-digit hexadecimal number (e.g. 0XFFFF for solid line, or 0X5555 for dotted lines).
- (**20 points bonus/extra part**) input one or multiple fill colors to color the given polygon if it is concave. To do this part, you need to implement the algorithm to identify a concave polygon first, and then convert it into multiple convex polygons (assume that user inputs simple polygons only).

Figures 1 and 2 show the menus available in the main form of the program allowing the user to open a window and change the settings of program. Keep in mind that any change in the settings will affect only the polygons drawn after the change.

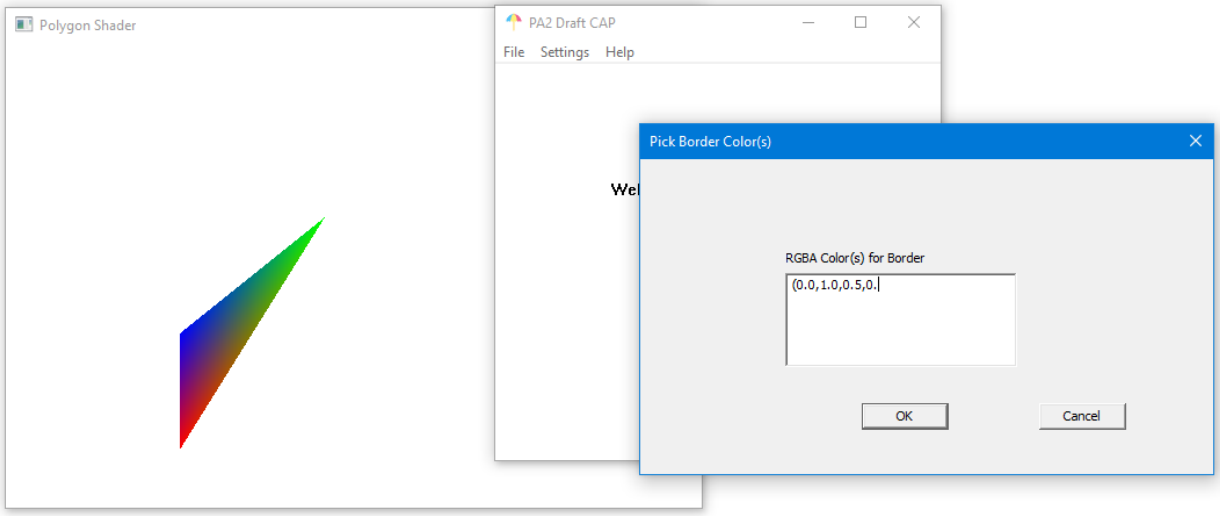

Figure 2: Entering a RGBA color for border

3 Deliverables

Submit a zip file containing the following items:

- A pdf file containing your answer to the exercises.
- The source files of your program for the programming assignments (.cpp files)
- A short mp4 video (less than 2 minutes) illustrating how to interact with the program to draw and color polygons.