CS 460
HW-1
Selin DINC

Problem Statement

drawLine File

- Drawing the letters of "WAKE" on the left side of the window automatically by using the Bresenham line drawing algorithm.
- Drawing the XYZ coordinate system in the middle of the window by using the Bresenham line drawing algorithm.
- Drawing the letters of "WAKE" on the right side of the window automatically by using the OpenGL's line drawing function called `GL_LINES`.
- Drawing line stipples with different widths on the left side of the window by using the midpoint line drawing algorithm.
- Drawing line stipples with different widths on the right side of the window by using the OpenGL's line drawing functions like `GL_LINES`, `GL_LINE_STIPPLE` and `glLineWidth`.

mouseClick File

- Drawing a series of connected line segments using the mouse anywhere in the window by using the OpenGL's line drawing and mouse functions. The user will click to create control points, and the program will draw the lines between the first point and the second, the second and third, the third and the fourth, etc.

Algorithm Design

drawLine File

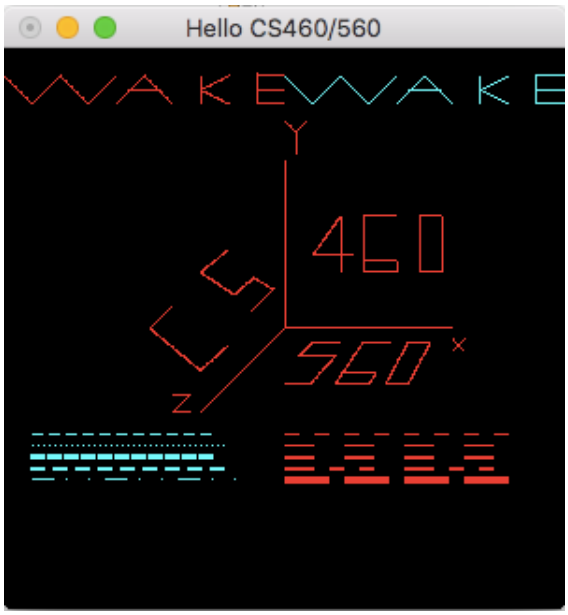

Figure 1: Output of the drawLine program

There are six functions for the drawLine program; bresenham, mid_point, setQuadrant, convertQuadrant, DisplayWakeOpenGL and display.

drawLine program creates a window that shows “WAKE” letters both on the top left and the right part of the window. The “WAKE” on the left side of the window is drawn by using the Bresenham Line drawing algorithm. The “WAKE” on the right side of the program is drawn by using the OpenGL’s functions. In the middle of the window program draws a XYZ coordinate system by using the Bresenham Line drawing algorithm. At the left bottom of the window program draws stipple lines with different widths by using Midpoint line drawing algorithm. Finally, at the bottom right of the window program draws again stipple lines but this time using the OpenGL’s functions.

- **bresenham** function applies the Bresenham Line Drawing algorithm.
- **mid_point** function applies the Midpoint Line Drawing a algorithm.
- **setQuadrant** function finds the line’s quadrant and sets the quadrant of the line (from I to VIII) by using a flag called whichQuad.
- **convertQuadrant** function checks the quadrant of the line and then converts the quadrant of the line to Quadrant I if it is already not there.
- **DisplayWakeOpenGL** function draws “WAKE” letters and stipple lines with different widths on the right side of the window by using the OpenGL’s glLineWidth, glColor3f, glVertex2f, glLineStipple and GL_LINES functions.
- **display** function draws “WAKE” letters and XYZ coordinate system by using bresenham algorithm and stipple lines with different widths by using mid_point algorithm.

mouseClick File

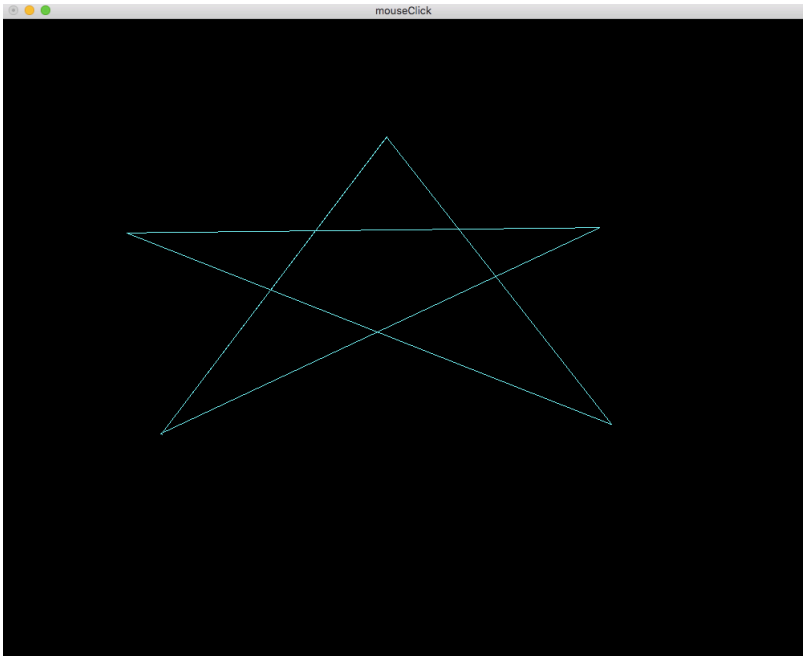

Figure 2: Output of the mouseClick program

mouseClick program creates a window that user can draw series of connected line segments using the mouse anywhere in the window by using the OpenGL’s line drawing and mouse functions. User can click to create control points, and the program will draw the lines between the first point and the second, the second and third, the third and the fourth, etc. By

hitting the left click button user adds a new control point. Then, moving the mouse without clicking any buttons draws a line from the previously added control point to the current cursor position. The right button adds the last control point. Program draws all the lines created up to this point until the user clicks the left button again. If the user clicks the left button again after the right button, then program clears all the lines drawn before.

mouseClick program consist of three functions; mouse, motion and display.

- **mouse** function sets the line points and then pushes to a vector that consist lines with each different mouse button click.
- **motion** function checks if the left button clicked and makes the motion possible by using glutPostRedispay function of openGL.
- **Display** function draws lines and display them.

How To Run

drawLine File

```
g++ main.cpp -o a.out -framework OpenGL -framework GLUT
```

mouseClick File

```
g++ main.cpp -o a.out -framework OpenGL -framework GLUT
```