B.Sc. Computer Science 3.2

ICS 2311 COMPUTER GRAPHICS

SCT211-0848/2018 - JANY MUONG
SCT211-0079/2022 - JORAM KIREKI
SCT211-0003/2022 - JOSPHAT WAWERU
SCT211-0535/2022 - AKECH ATEM

SUBMITTED: **April 6th, 2025**

G7

ICS 2311: Computer Graphics – OpenGL Groupwork

CAT – WRITE UP:

mail to: jkuatnotes7@gmail.com
subject: GROUP 7: ICS 2311

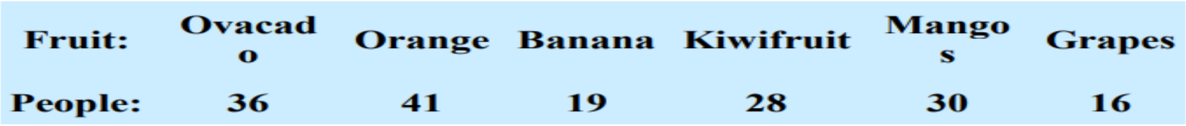
Background Context:

Group work using OpenGL/GLUT/GLEW.
Working on question 7 from the psets

Information:

This write-up contains our solutions for questions 7, captured in screenshots and the code for each part (**a** and **b**) of the question are at the end of the screenshots. They can be used to reconstruct the OpenGL/C code if opening the zip files fails. The same code is also in the zip archive.

Question 7:
A survey was carried out in Gachororo about youth preference on fruits. 150 youth were interviewed about their fruits of preference as follows:



Fruit:	Ovacado	Orange	Banana	Kiwifruit	Mangos	Grapes
People:	36	41	19	28	30	16

QUESTION 7 – Solution

We are being tested for these core concepts:

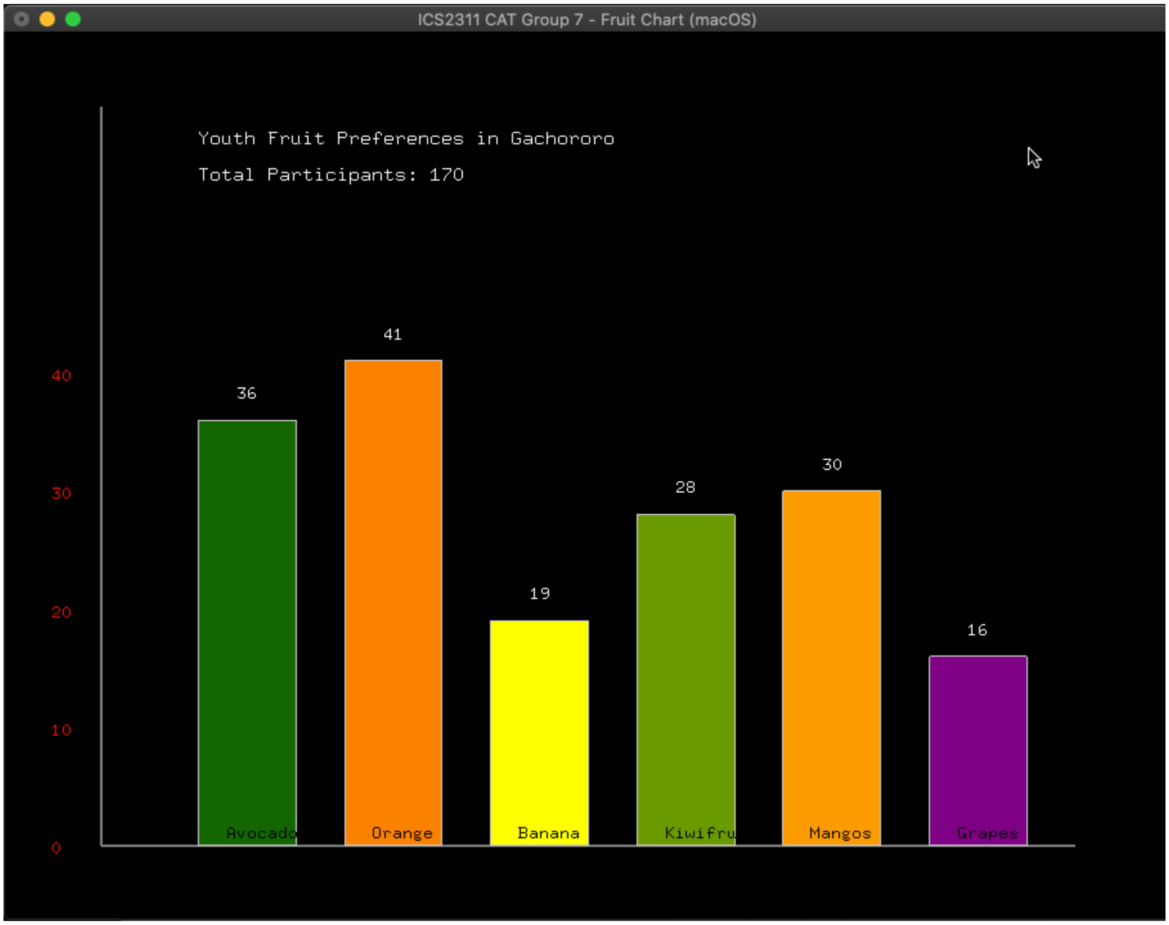
1. **OpenGL:** create basic 2D visualizations using OpenGL, **coordinate systems** and transformations, output primitives (lines, rectangles, text)
2. **Miscellaneous**: creating accurate bar charts from raw data, proper **scaling** of data to fit display window, effective labeling of chart elements
3. **Solution**:
   - **Part (a)**: scaling of data points to fill display area, color mapping (bars colored to match actual fruits), axis labeling with specific color requirements (x-axis black, y-axis red)
   - **Part (b)**: coordinate system manipulation, translate/offset graphical elements, maintaining all chart functionality while changing origin point

G7

a) First Part: Response/OpenGL:

```
➜ gachororo_fruit
clang -framework OpenGL -framework GLUT a_fruit_bars.c -o a_fruit_bars
➜ gachororo_fruit
./a_fruit_bars
```

OpenGL Display



ICS2311 CAT Group 7 - Fruit Chart (macOS)

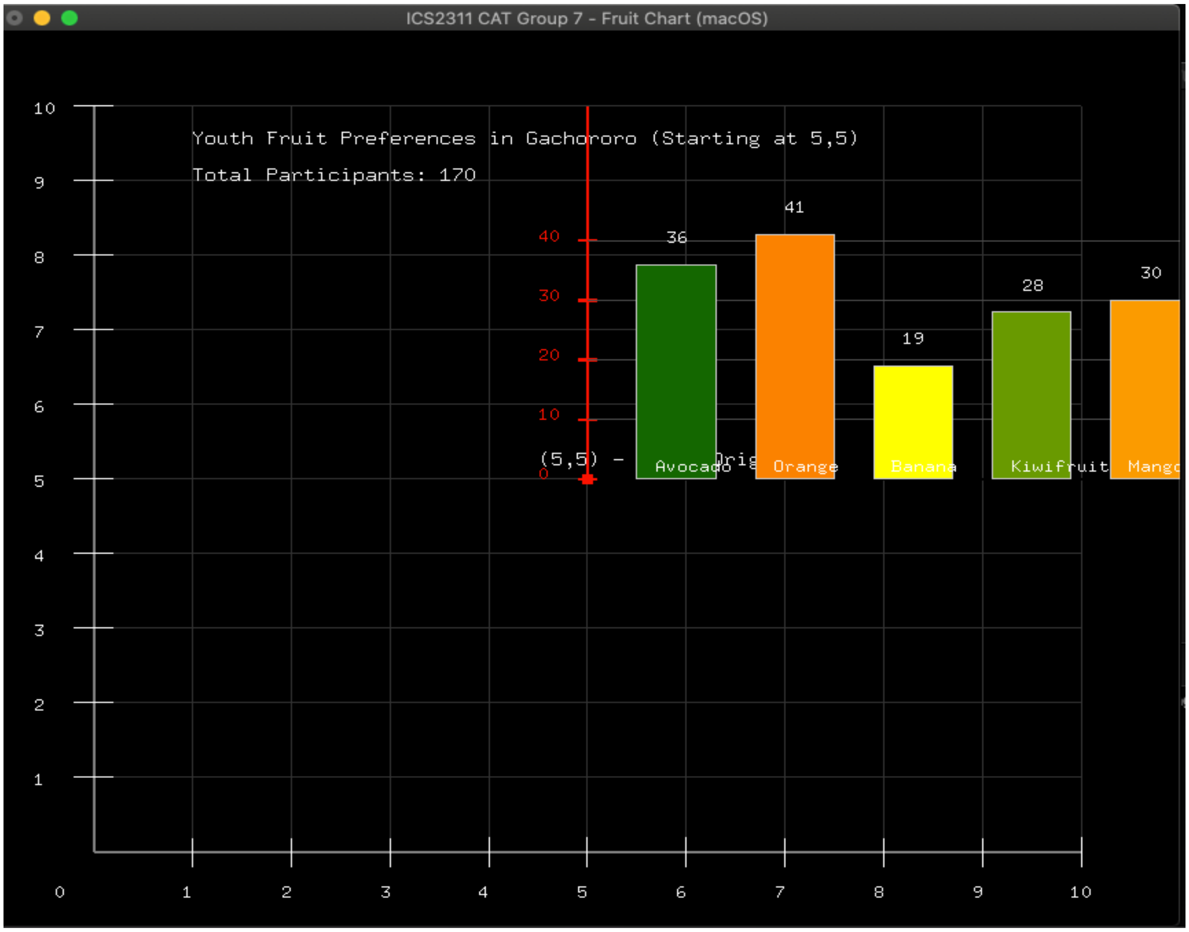
Youth Fruit Preferences in Gachororo

Total Participants: 170

b) Second Part: Response/OpenGL:

This is meant to show transformations ie translation that starts at origin (5, 5). We have included a second coordinate systems for comparison.

```
➜ gachororo_fruit
clang -framework OpenGL -framework GLUT b_fruit_bars.c -o b_fruit_bars
➜ gachororo_fruit
./b_fruit_bars
```



G7

OpenGL/Code:

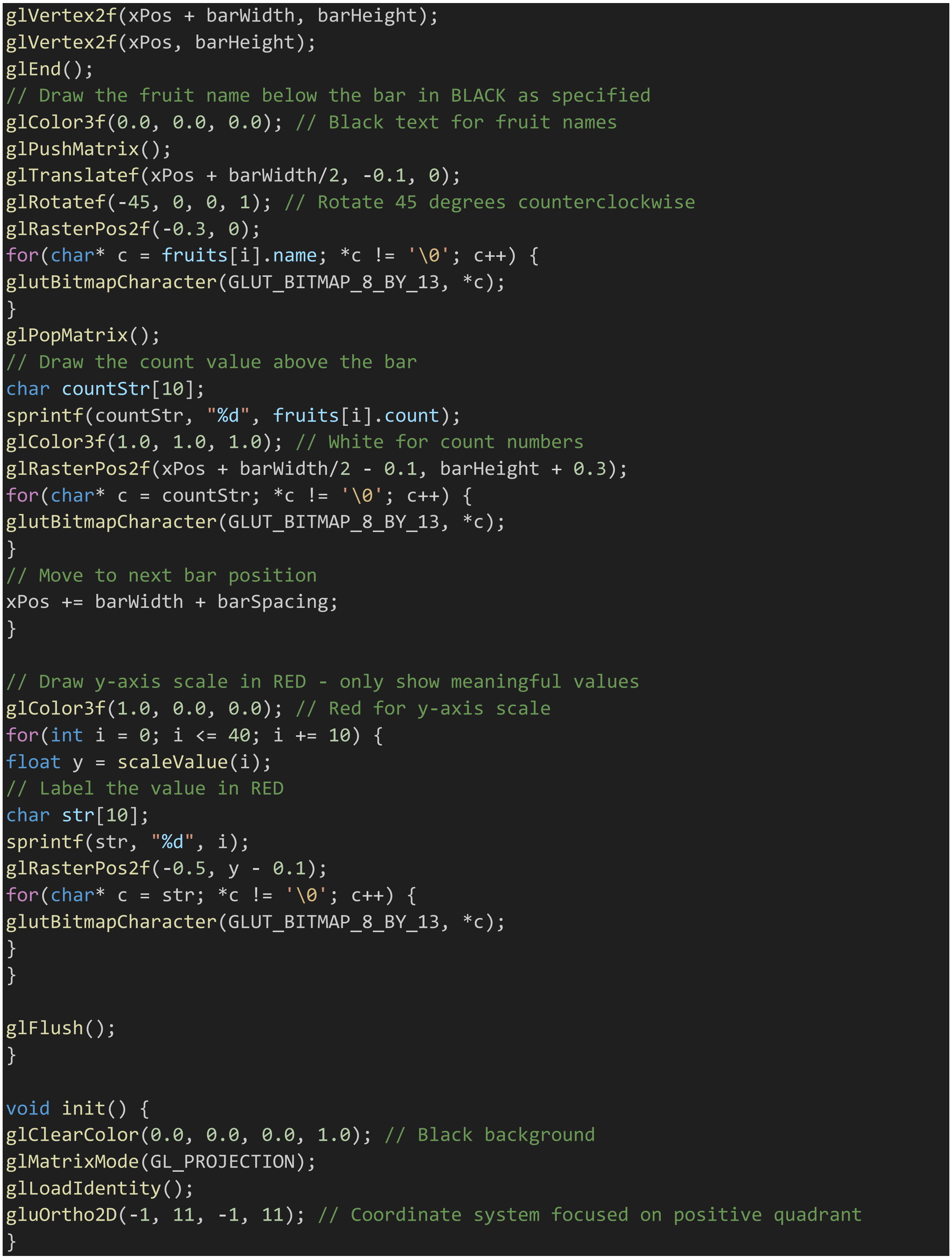
Part A:

```
// ICS2311 CAT Group 7 - Fruit Preference Bar Chart (macOS Version)
// Compile with: clang -framework OpenGL -framework GLUT a_fruit_bars.c -o FruitBars

#include <OpenGL/gl.h>
#include <GLUT/glut.h>
#include <stdio.h>
#include <math.h>
#include <string.h>

// Structure to hold fruit data
typedef struct {
char name[20];
int count;
float color[3]; // RGB values
} FruitData;

// Global array of fruit data
FruitData fruits[] = {
{"Avocado", 36, {0.0, 0.4, 0.0}}, // Dark green
{"Orange", 41, {1.0, 0.5, 0.0}}, // Orange color
{"Banana", 19, {1.0, 1.0, 0.0}}, // Yellow
{"Kiwifruit", 28, {0.4, 0.6, 0.0}}, // Kiwi green
{"Mangos", 30, {1.0, 0.6, 0.2}}, // Mango orange
{"Grapes", 16, {0.5, 0.0, 0.5}} // Purple
};

const int numFruits = sizeof(fruits) / sizeof(fruits[0]);
const int maxCount = 50; // Max value for y-axis (slightly more than our max data point)
const float barWidth = 1.0;
const float barSpacing = 0.5;

void drawCoordinateSystem() {
// Draw main axes only
glColor3f(0.5, 0.5, 0.5); // Gray color for axes
glLineWidth(2.0);
glBegin(GL_LINES);
// x-axis
glVertex2f(0.0, 0.0);
glVertex2f(10.0, 0.0);
// y-axis
glVertex2f(0.0, 0.0);
glVertex2f(0.0, 10.0);
glEnd();
}
```

G7

```
// Function to draw text in the scene
void drawText(const char* text, float x, float y) {
glRasterPos2f(x, y);
for(const char* c = text; *c != '\0'; c++) {
glutBitmapCharacter(GLUT_BITMAP_9_BY_15, *c);
}
}

// Function to scale fruit counts to fit in the display window
float scaleValue(int value) {
// Scale to range 0 to 8 (to fit in our coordinate system)
return (float)value / maxCount * 8.0;
}

void display() {
glClear(GL_COLOR_BUFFER_BIT);
// Draw coordinate system (just the axes)
drawCoordinateSystem();

// Draw chart title
glColor3f(1.0, 1.0, 1.0); // White text
drawText("Youth Fruit Preferences in Gachororo", 1.0, 9.5);
drawText("Total Participants: 170", 1.0, 9.0);
// Draw x-axis label in BLACK as specified
glColor3f(0.0, 0.0, 0.0); // Black text for x-axis
drawText("Fruit Type", 8.0, 0.5);
// Starting position for first bar (in positive quadrant)
float xPos = 1.0;
// Draw bars
for(int i = 0; i < numFruits; i++) {
// Calculate bar height based on data
float barHeight = scaleValue(fruits[i].count);
// Draw the bar with fruit color
glColor3fv(fruits[i].color);
glBegin(GL_QUADS);
glVertex2f(xPos, 0.0);
glVertex2f(xPos + barWidth, 0.0);
glVertex2f(xPos + barWidth, barHeight);
glVertex2f(xPos, barHeight);
glEnd();
// Draw bar outline
glColor3f(0.8, 0.8, 0.8); // Light gray outline
glLineWidth(1.0);
glBegin(GL_LINE_LOOP);
glVertex2f(xPos, 0.0);
glVertex2f(xPos + barWidth, 0.0);
```

G7



```
glVertex2f(xPos + barWidth, barHeight);
glVertex2f(xPos, barHeight);
glEnd();
// Draw the fruit name below the bar in BLACK as specified
glColor3f(0.0, 0.0, 0.0); // Black text for fruit names
glPushMatrix();
glTranslatef(xPos + barWidth/2, -0.1, 0);
glRotatef(-45, 0, 0, 1); // Rotate 45 degrees counterclockwise
glRasterPos2f(-0.3, 0);
for(char* c = fruits[i].name; *c != '\0'; c++) {
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
}
glPopMatrix();
// Draw the count value above the bar
char countStr[10];
sprintf(countStr, "%d", fruits[i].count);
glColor3f(1.0, 1.0, 1.0); // White for count numbers
glRasterPos2f(xPos + barWidth/2 - 0.1, barHeight + 0.3);
for(char* c = countStr; *c != '\0'; c++) {
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
}
// Move to next bar position
xPos += barWidth + barSpacing;
}

// Draw y-axis scale in RED - only show meaningful values
glColor3f(1.0, 0.0, 0.0); // Red for y-axis scale
for(int i = 0; i <= 40; i += 10) {
float y = scaleValue(i);
// Label the value in RED
char str[10];
sprintf(str, "%d", i);
glRasterPos2f(-0.5, y - 0.1);
for(char* c = str; *c != '\0'; c++) {
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
}
}

glFlush();
}

void init() {
glClearColor(0.0, 0.0, 0.0, 1.0); // Black background
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
gluOrtho2D(-1, 11, -1, 11); // Coordinate system focused on positive quadrant
}
```

G7

```
int main(int argc, char** argv) {
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(800, 800);
glutInitWindowPosition(100, 100);
glutCreateWindow("ICS2311 CAT Group 7 - Fruit Chart (macOS)");

init();
glutDisplayFunc(display);
glutMainLoop();
return 0;
}
```

Part B:

```
// ICS2311 CAT Group 7 - Fruit Preference Bar Chart (macOS Version)
// Compile with: clang -framework OpenGL -framework GLUT b_fruit_bars.c -o FruitBars

#include <OpenGL/gl.h>
#include <GLUT/glut.h>
#include <stdio.h>
#include <math.h>
#include <string.h>

// Structure to hold fruit data
typedef struct {
char name[20];
int count;
float color[3]; // RGB values
} FruitData;

// Global array of fruit data
FruitData fruits[] = {
{"Avocado", 36, {0.0, 0.4, 0.0}}, // Dark green
{"Orange", 41, {1.0, 0.5, 0.0}}, // Orange color
{"Banana", 19, {1.0, 1.0, 0.0}}, // Yellow
{"Kiwifruit", 28, {0.4, 0.6, 0.0}}, // Kiwi green
{"Mangos", 30, {1.0, 0.6, 0.2}}, // Mango orange
{"Grapes", 16, {0.5, 0.0, 0.5}} // Purple
};
```

G7

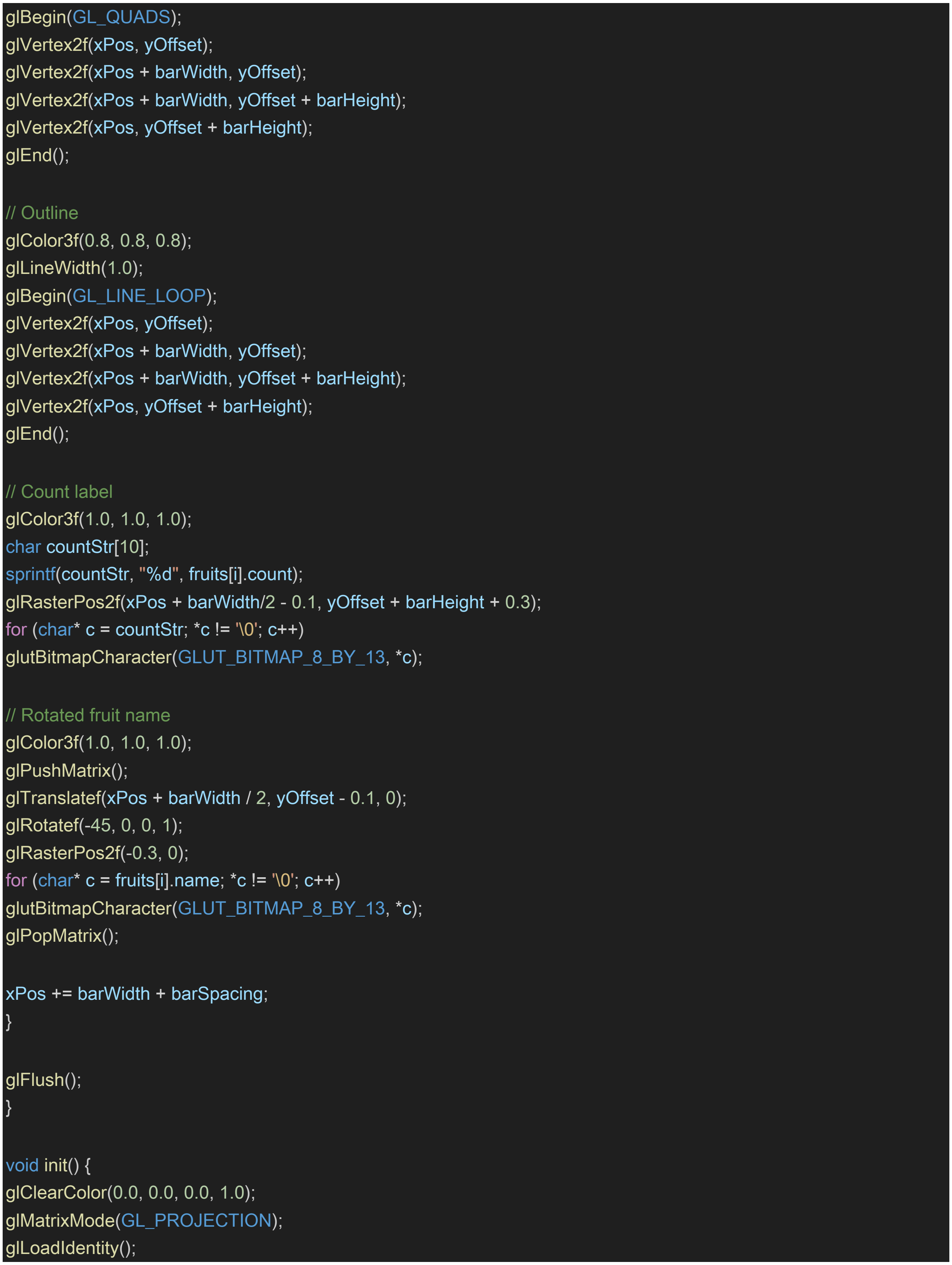
```
const int numFruits = sizeof(fruits) / sizeof(fruits[0]);
const int maxCount = 50;
const float barWidth = 0.8;
const float barSpacing = 0.4;

// Offset for the chart (starting point at (5,5))
const float xOffset = 5.0;
const float yOffset = 5.0;

// Function to draw text in the scene
void drawText(const char* text, float x, float y) {
glRasterPos2f(x, y);
for(const char* c = text; *c != '\0'; c++) {
glutBitmapCharacter(GLUT_BITMAP_9_BY_15, *c);
}
}

void drawCoordinateSystem() {
// Draw main axes
glColor3f(0.5, 0.5, 0.5);
glLineWidth(2.0);
glBegin(GL_LINES);
glVertex2f(0.0, 0.0); glVertex2f(10.0, 0.0);
glVertex2f(0.0, 0.0); glVertex2f(0.0, 10.0);
glEnd();

// Grid lines
glColor3f(0.2, 0.2, 0.2);
glLineWidth(1.0);
glBegin(GL_LINES);
for (int i = 1; i <= 10; i++) {
glVertex2f(i, 0.0); glVertex2f(i, 10.0);
glVertex2f(0.0, i); glVertex2f(10.0, i);
}
glEnd();

// Ticks and numbers
glColor3f(1.0, 1.0, 1.0);
for (int i = 1; i <= 10; i++) {
glBegin(GL_LINES);
glVertex2f(i, -0.2); glVertex2f(i, 0.2);
glVertex2f(-0.2, i); glVertex2f(0.2, i);
glEnd();

char str[10];
```

G7

```
sprintf(str, "%d", i);
glRasterPos2f(i - 0.1, -0.6);
for (char* c = str; *c != '\0'; c++)
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);

glRasterPos2f(-0.6, i - 0.1);
for (char* c = str; *c != '\0'; c++)
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
}

// Origin label
glRasterPos2f(-0.4, -0.6);
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, '0');

// Highlight (5,5)
glColor3f(1.0, 0.0, 0.0);
glPointSize(8.0);
glBegin(GL_POINTS);
glVertex2f(xOffset, yOffset);
glEnd();

glColor3f(1.0, 1.0, 1.0);
drawText("(5,5) - Chart Origin", xOffset - 0.5, yOffset + 0.2);
}

float scaleValue(int value) {
return (float)value / maxCount * 4.0;
}

void drawChartAxes() {
glLineWidth(2.5);
// x-axis - BLACK
glColor3f(0.0, 0.0, 0.0);
glBegin(GL_LINES);
glVertex2f(xOffset, yOffset);
glVertex2f(xOffset + 8.0, yOffset);
glEnd();
// y-axis - RED
glColor3f(1.0, 0.0, 0.0);
glBegin(GL_LINES);
glVertex2f(xOffset, yOffset);
glVertex2f(xOffset, yOffset + 5.0);
glEnd();
// Ticks on y-axis
glColor3f(1.0, 0.0, 0.0);
```

```
for (int i = 0; i <= 40; i += 10) {
float y = yOffset + scaleValue(i);
glBegin(GL_LINES);
glVertex2f(xOffset - 0.1, y);
glVertex2f(xOffset + 0.1, y);
glEnd();

char str[10];
sprintf(str, "%d", i);
glRasterPos2f(xOffset - 0.5, y);
for (char* c = str; *c != '\0'; c++)
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
}

// Horizontal grid lines
glColor3f(0.3, 0.3, 0.3);
glLineWidth(1.0);
for (int i = 10; i <= 40; i += 10) {
float y = yOffset + scaleValue(i);
glBegin(GL_LINES);
glVertex2f(xOffset, y);
glVertex2f(xOffset + 8.0, y);
glEnd();
}
}

void display() {
glClear(GL_COLOR_BUFFER_BIT);

drawCoordinateSystem();
drawChartAxes();

glColor3f(1.0, 1.0, 1.0);
drawText("Youth Fruit Preferences in Gachororo (Starting at 5,5)", 1.0, 9.5);
drawText("Total Participants: 170", 1.0, 9.0);

// X-axis label
glColor3f(0.0, 0.0, 0.0);
drawText("Fruit Type", xOffset + 6.0, yOffset - 0.5);

float xPos = xOffset + 0.5;
for (int i = 0; i < numFruits; i++) {
float barHeight = scaleValue(fruits[i].count);

glColor3fv(fruits[i].color);
```

G7



```
glBegin(GL_QUADS);
glVertex2f(xPos, yOffset);
glVertex2f(xPos + barWidth, yOffset);
glVertex2f(xPos + barWidth, yOffset + barHeight);
glVertex2f(xPos, yOffset + barHeight);
glEnd();

// Outline
glColor3f(0.8, 0.8, 0.8);
glLineWidth(1.0);
glBegin(GL_LINE_LOOP);
glVertex2f(xPos, yOffset);
glVertex2f(xPos + barWidth, yOffset);
glVertex2f(xPos + barWidth, yOffset + barHeight);
glVertex2f(xPos, yOffset + barHeight);
glEnd();

// Count label
glColor3f(1.0, 1.0, 1.0);
char countStr[10];
sprintf(countStr, "%d", fruits[i].count);
glRasterPos2f(xPos + barWidth/2 - 0.1, yOffset + barHeight + 0.3);
for (char* c = countStr; *c != '\0'; c++)
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);

// Rotated fruit name
glColor3f(1.0, 1.0, 1.0);
glPushMatrix();
glTranslatef(xPos + barWidth / 2, yOffset - 0.1, 0);
glRotatef(-45, 0, 0, 1);
glRasterPos2f(-0.3, 0);
for (char* c = fruits[i].name; *c != '\0'; c++)
glutBitmapCharacter(GLUT_BITMAP_8_BY_13, *c);
glPopMatrix();

xPos += barWidth + barSpacing;
}

glFlush();
}

void init() {
glClearColor(0.0, 0.0, 0.0, 1.0);
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
```

```
gluOrtho2D(-1, 11, -1, 11);
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
}

int main(int argc, char** argv) {
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(800, 800);
glutInitWindowPosition(100, 100);
glutCreateWindow("ICS2311 CAT Group 7 - Fruit Chart (macOS)");
init();
glutDisplayFunc(display);
glutMainLoop();
return 0;
}
```

OPENGL/ PYTHON CODE:

PART A: a_fruit_bars.py

```
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
import sys

fruits = [
{"name": "Avocado", "count": 36, "color": [0.34, 0.51, 0.01]},
{"name": "Orange", "count": 41, "color": [1.00, 0.65, 0.00]},
{"name": "Banana", "count": 19, "color": [0.89, 0.81, 0.34]},
{"name": "Kiwifruit", "count": 28, "color": [0.62, 0.89, 0.31]},
{"name": "Mangos", "count": 30, "color": [1.00, 0.78, 0.25]},
{"name": "Grapes", "count": 16, "color": [0.50, 0.19, 0.58]}
]

num_fruits = len(fruits)
window_width, window_height = 800, 600

def reshape(width, height):
global window_width, window_height
window_width, window_height = width, height
glViewport(0, 0, width, height)
glutPostRedisplay()
```

G7

```
def draw_text(text, x, y, r, g, b):
glColor3f(r, g, b)
glRasterPos2f(x, y)
for c in text:
glutBitmapCharacter(GLUT_BITMAP_9_BY_15, ord(c))

def display():
glClear(GL_COLOR_BUFFER_BIT)
glMatrixMode(GL_PROJECTION)
glLoadIdentity()
gluOrtho2D(0, window_width, 0, window_height)

glMatrixMode(GL_MODELVIEW)
glLoadIdentity()

# Title and subtitle
draw_text("Youth Fruit Preferences in Gachororo", window_width / 2 - 180, window_height - 30, 0.0, 0.0, 0.0)
draw_text("Total Participants: 170", window_width / 2 - 100, window_height - 60, 0.0, 0.0, 0.0)

# Axes
glColor3f(1.0, 0.0, 0.0) # Red Y-axis
glLineWidth(2.0)
glBegin(GL_LINES)
glVertex2f(50.0, 50.0)
glVertex2f(50.0, window_height - 50.0)
glEnd()

glColor3f(0.0, 0.0, 0.0) # Black X-axis
glBegin(GL_LINES)
glVertex2f(50.0, 50.0)
glVertex2f(window_width - 50.0, 50.0)
glEnd()

draw_text("Fruit Type", window_width / 2 - 40, 20, 0.0, 0.0, 0.0)

# Bar settings
bar_width = (window_width - 150) / num_fruits * 0.7
bar_spacing = (window_width - 150) / num_fruits * 0.3
max_bar_height = window_height - 150
start_x = 100.0

# Draw bars
for fruit in fruits:
bar_height = (fruit["count"] / 50) * max_bar_height
```

G7

```
glColor3fv(fruit["color"])
glBegin(GL_QUADS)
glVertex2f(start_x, 50.0)
glVertex2f(start_x + bar_width, 50.0)
glVertex2f(start_x + bar_width, 50.0 + bar_height)
glVertex2f(start_x, 50.0 + bar_height)
glEnd()

# Draw value
draw_text(str(fruit["count"]), start_x + bar_width / 2 - 10, 60 + bar_height, 0.0, 0.0, 0.0)

# Draw fruit name (rotated)
glPushMatrix()
glTranslatef(start_x + bar_width / 2, 30, 0)
glRotatef(45, 0, 0, 1)
draw_text(fruit["name"], 0, 0, 0.0, 0.0, 0.0)
glPopMatrix()

start_x += bar_width + bar_spacing

# Draw y-axis labels
for i in range(0, 51, 10):
y = 50 + (i / 50) * max_bar_height
glColor3f(1.0, 0.0, 0.0)
glBegin(GL_LINES)
glVertex2f(45, y)
glVertex2f(50, y)
glEnd()
draw_text(str(i), 30, y - 5, 1.0, 0.0, 0.0)

glFlush()

def init():
glClearColor(1.0, 1.0, 1.0, 1.0)

def main():
glutInit(sys.argv)
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB)
glutInitWindowSize(window_width, window_height)
glutCreateWindow(b"ICS2311 Group 7: Fruit Preference Survey")
glutReshapeFunc(reshape)
glutDisplayFunc(display)
init()
glutMainLoop()
```

```
if __name__ == "__main__":
main()
```

PART B: b_fruit_bars.py

```
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
import sys

# Fruit data
fruits = [
{"name": "Avocado", "count": 36, "color": [0.34, 0.51, 0.01]}, # dark green
{"name": "Orange", "count": 41, "color": [1.00, 0.65, 0.00]}, # orange
{"name": "Banana", "count": 19, "color": [0.89, 0.81, 0.34]}, # yellow
{"name": "Kiwifruit", "count": 28, "color": [0.62, 0.89, 0.31]}, # kiwi green
{"name": "Mangos", "count": 30, "color": [1.00, 0.78, 0.25]}, # mango orange
{"name": "Grapes", "count": 16, "color": [0.50, 0.19, 0.58]} # purple
]

num_fruits = len(fruits)
window_width, window_height = 800, 600

# Offset for the chart (starting point at (5,5))
x_offset = 5.0
y_offset = 5.0

def reshape(width, height):
global window_width, window_height
window_width, window_height = width, height
glViewport(0, 0, width, height)
glutPostRedisplay()

def draw_text(text, x, y, r, g, b):
glColor3f(r, g, b)
glRasterPos2f(x, y)
for c in text:
glutBitmapCharacter(GLUT_BITMAP_9_BY_15, ord(c))

def draw_base_coordinate_system():
glColor3f(0.5, 0.5, 0.5)
glLineWidth(1.0)
```

G7

```
glBegin(GL_LINES)
glVertex2f(0.0, 0.0); glVertex2f(12.0, 0.0) # x-axis
glVertex2f(0.0, 0.0); glVertex2f(0.0, 12.0) # y-axis
glEnd()

draw_text("0", -0.3, -0.3, 0.5, 0.5, 0.5)
draw_text("X", 12.2, 0.0, 0.5, 0.5, 0.5)
draw_text("Y", 0.0, 12.2, 0.5, 0.5, 0.5)
draw_text("World Coordinate System", 2.0, 11.5, 0.5, 0.5, 0.5)

glColor3f(0.3, 0.3, 0.3)
glLineWidth(0.5)
glBegin(GL_LINES)
for i in range(1, 13):
glVertex2f(i, 0.0); glVertex2f(i, 12.0)
glVertex2f(0.0, i); glVertex2f(12.0, i)
glEnd()

for i in range(1, 13):
if i % 2 == 0 or i == 5:
draw_text(str(i), i - 0.1, -0.3, 0.5, 0.5, 0.5)
draw_text(str(i), -0.3, i - 0.1, 0.5, 0.5, 0.5)

glColor3f(1.0, 0.0, 0.0)
glPointSize(8.0)
glBegin(GL_POINTS)
glVertex2f(x_offset, y_offset)
glEnd()

glColor3f(1.0, 0.0, 0.0)
glLineWidth(1.5)
glBegin(GL_LINES)
glVertex2f(x_offset + 0.1, y_offset + 0.1)
glVertex2f(x_offset + 0.2, y_offset + 0.2)
glEnd()

def draw_chart_axes():
glColor3f(1.0, 0.0, 0.0)
glLineWidth(2.0)
glBegin(GL_LINES)
glVertex2f(x_offset, y_offset)
glVertex2f(x_offset, y_offset + 5.0)
glEnd()

glColor3f(0.0, 0.0, 0.0)
```

G7

```
glBegin(GL_LINES)
glVertex2f(x_offset, y_offset)
glVertex2f(x_offset + 8.0, y_offset)
glEnd()

draw_text("Chart X-Axis", x_offset + 8.2, y_offset, 0.0, 0.0, 0.0)
draw_text("Chart Y-Axis", x_offset - 0.5, y_offset + 5.2, 1.0, 0.0, 0.0)

glColor3f(1.0, 0.0, 0.0)
for i in range(0, 51, 10):
y = y_offset + (i / 50.0) * 5.0
glBegin(GL_LINES)
glVertex2f(x_offset - 0.1, y)
glVertex2f(x_offset, y)
glEnd()
draw_text(str(i), x_offset - 0.5, y - 0.1, 1.0, 0.0, 0.0)

def display():
glClear(GL_COLOR_BUFFER_BIT)

draw_base_coordinate_system()
draw_chart_axes()

draw_text("Youth Fruit Preferences in Gachororo (Starting at 5,5)", 2.0, 11.0, 1.0, 1.0, 1.0)
draw_text("Total Participants: 170", 2.0, 10.5, 1.0, 1.0, 1.0)
draw_text("Fruit Type", x_offset + 4.0, y_offset - 0.5, 0.0, 0.0, 0.0)

bar_width = 0.8
bar_spacing = 0.4
max_bar_height = 5.0
start_x = x_offset + 0.5

for fruit in fruits:
bar_height = (fruit["count"] / 50.0) * max_bar_height
if bar_height < 0.1:
bar_height = 0.1

glColor3fv(fruit["color"])
glBegin(GL_QUADS)
glVertex2f(start_x, y_offset)
glVertex2f(start_x + bar_width, y_offset)
glVertex2f(start_x + bar_width, y_offset + bar_height)
glVertex2f(start_x, y_offset + bar_height)
glEnd()
```

G7

```
glColor3f(0.8, 0.8, 0.8)
glLineWidth(1.0)
glBegin(GL_LINE_LOOP)
glVertex2f(start_x, y_offset)
glVertex2f(start_x + bar_width, y_offset)
glVertex2f(start_x + bar_width, y_offset + bar_height)
glVertex2f(start_x, y_offset + bar_height)
glEnd()

draw_text(str(fruit["count"]), start_x + bar_width / 2 - 0.2, y_offset + bar_height + 0.2, 0.0, 0.0, 0.0)

glPushMatrix()
glTranslatef(start_x + bar_width / 2, y_offset - 0.2, 0)
glRotatef(-45, 0, 0, 1)
draw_text(fruit["name"], 0, 0, 1.0, 1.0, 1.0)
glPopMatrix()

start_x += bar_width + bar_spacing

glFlush()

def init():
glClearColor(0.0, 0.0, 0.0, 1.0)
glMatrixMode(GL_PROJECTION)
glLoadIdentity()
gluOrtho2D(-1, 14, -1, 14)
glMatrixMode(GL_MODELVIEW)
glLoadIdentity()

def main():
glutInit(sys.argv)
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB)
glutInitWindowSize(window_width, window_height)
glutInitWindowPosition(100, 100)
glutCreateWindow(b"ICS2311 Group 7: Fruit Preference Survey (Starting at 5,5)")
init()
glutDisplayFunc(display)
glutReshapeFunc(reshape)
glutMainLoop()

if __name__ == "__main__":
main()
```

G7

END FILE