Computer Graphics Lab Final Project

Team members:

Md. Tushar Ahmed ---------- 201-15-3549 ---------- 55_PC-D1

Rifah Tasnia ---------- 201-15-3571 ---------- 55_PC-D2

Jawad Jarif Khan ---------- 201-15-3725 ---------- 55_PC-D2

Md Shariful Islam ---------- 201-15-3301 ---------- 55_PC- D1

Rahul Malo Romen ---------- 201-15-3456 ----------55_PC-D2

Project Title: DIU Campus (Day mode + Night mode)

Code:

```
#include <GL/gl.h>
#include <GL/glut.h>
#include <math.h>
#include <iostream>
#include <vector>
#include <windows.h>


void init(void)
{
   glClearColor(0.0,1.0,1.0,1.0); //GLfloat red,green,blue,alpha initial value 0 alpha values used by glclear to clear the color buffers
   glMatrixMode(GL_PROJECTION);  // To specify which matrix is the current matrix & projection applies subsequent matrix to projecton matrix stack
   glLoadIdentity();
   glOrtho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0);
}
```

```
///=====================================///
///         Circle Function           ///
///=====================================///


void circle(GLfloat radius, GLfloat cx, GLfloat cy)
{
   glBegin(GL_POLYGON);
   glColor3f(0.0, 0.0, 0.0);
   glVertex2f(cx, cy);
   for (int i = 0; i <= 360; i++)
   {
      float angle = i * 3.1416 / 180;
      float x = radius * cos(angle);
      float y = radius * sin(angle);
      glVertex2f((x + cx), (y + cy));
   }
   glEnd();
}

///=====================================///
///         Sun Function              ///
///=====================================///
```

```
float sunX = 0.125;
bool isDay = true;
void drawSun(float centerX, float centerY, float radius)
{
   // Set the sun color based on day/night transition
   if (isDay)
   {
      glColor3f(1.0, 0.9, 0.1); // Yellow color
   }
   else
   {
      glColor3f(1.0, 1.0, 1.0); // White color (moon)
   }

   // Draw the outer circle
   glBegin(GL_POLYGON);
   glVertex2f(centerX, centerY); // Center point

   int numSegments = 100;
   for (int i = 0; i <= numSegments; i++)
   {
      float theta = 2.0 * 3.14159 * float(i) / float(numSegments);
      float x = centerX + radius * cos(theta);
      float y = centerY + radius * sin(theta);
      glVertex2f(x, y);
   }
```

```
    glEnd();
}

///====================================///
///  BackGround Color Change Function    ///
///====================================///

void drawBackground()
{

    glBegin(GL_QUADS);
    if (isDay)
    {
        glColor3f(0.4, 0.6, 0.8); // Deep blue color
    }
    else
    {
        glColor3f(0.584, 0.792, 0.645); // Evening orange color

    }
    glVertex3f(0, 0.375f, 0.0f);
    glVertex3f(1, 0.375f, 0.0f);

    if (isDay)
    {
        glColor3f(0.984, 0.792, 0.545); // Evening orange color
    }
```

```
    else
    {
        glColor3f(0.05, 0.05, 0.15); // Dark blue color (night)

    }
    glVertex3f(1, 1, 0.0f);
    glVertex3f(0, 1, 0.0f);
    glEnd();
}

/// Day or Night Sun Move
void update(int value)
{
    // Update the sun position
    sunX += 0.001;
    // Wrap the sun position to create a continuous animation
    if (sunX > 1.0)
    {
        sunX = -0.001;
        isDay = !isDay; // Toggle day/night transition
    }

    // Set the next update interval
    glutTimerFunc(10, update, 0);

    // Redraw the scene
    glutPostRedisplay();
```

```
}

/// Bus
float bx = 0;
void bus()
{
   glPushMatrix();
   glTranslatef(bx, 0, 0);

///.....................

   //bus
   glColor3f (1.0, 1.0, 0.0);
   glBegin(GL_QUADS);
   //Trapezoid
   glVertex3f(0.6f, 0.3375f, 0.0f);
   glVertex3f(0.8f, 0.3375f, 0.0f);
   glVertex3f(0.8f, 0.3875f, 0.0f);
   glVertex3f(0.625f,0.3875f, 0.0f);
   glEnd();

   //bus window
   glColor3f (1.0, 0.0, 0.0);
   glBegin(GL_QUADS);
   //Trapezoid
   glVertex3f(0.675f, 0.35f, 0.0f);
   glVertex3f(0.775f, 0.35f, 0.0f);
```

```
    glVertex3f(0.775f, 0.375f, 0.0f);
    glVertex3f(0.675f,0.375f, 0.0f);
    glEnd();

    //bus door
    glColor3f (0.0, 1.0, 1.0);
    glBegin(GL_QUADS);
    //Trapezoid
    glVertex3f(0.6375f, 0.3375f, 0.0f);
    glVertex3f(0.6625f, 0.3375f, 0.0f);
    glVertex3f(0.6625f, 0.375f, 0.0f);
    glVertex3f(0.6375f,0.375f, 0.0f);
    glEnd();

    //chaka
    circle(0.01,0.6875,0.3375);
    circle(0.01,0.75,0.3375);

///.....

    glPopMatrix();
    bx -= 0.00002;
    if (bx > 1)
       bx = 0.9;

    glutPostRedisplay();
```

```
}


///==================================///
///         Main Code              ///
///==================================//
void Draw()
{
   glClear(GL_COLOR_BUFFER_BIT);
   //our code start from here

   glColor3f (0, 0, 0);
   glBegin(GL_QUADS);
   //Trapezoid
   glVertex3f(0, 0.375f, 0.0f);
   glVertex3f(1, 0.375f, 0.0f);
   glVertex3f(1, 1, 0.0f);
   glVertex3f(0, 1, 0.0f);
   glEnd();


   ///Sun
   drawBackground();
   drawSun(sunX, 0.8125,0.0625);

   // AB-1 building
   // 1
```

```
glColor3f (0.67, 0.18, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.1875f, 0.375f, 0.0f);
glVertex3f(0.1875f, 0.5625f, 0.0f);
glVertex3f(0.275f, 0.5625f, 0.0f);
glVertex3f(0.275f, 0.375f, 0.0f);
glEnd();

glColor3f (0.40, 0.20, 0.40);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.2f, 0.375f, 0.0f);
glVertex3f(0.2f, 0.55f, 0.0f);
glVertex3f(0.2625f, 0.55f, 0.0f);
glVertex3f(0.2625f, 0.375f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.2125f, 0.5125f, 0.0f);
glVertex3f(0.2125f, 0.5375f, 0.0f);
glVertex3f(0.25f, 0.5375f, 0.0f);
```

```
glVertex3f(0.25f, 0.5125f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.2125f, 0.4625f, 0.0f);
glVertex3f(0.2125f, 0.4875f, 0.0f);
glVertex3f(0.25f, 0.4875f, 0.0f);
glVertex3f(0.25f, 0.4625f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.2125f, 0.4125f, 0.0f);
glVertex3f(0.2125f, 0.4375f, 0.0f);
glVertex3f(0.25f, 0.4375f, 0.0f);
glVertex3f(0.25f, 0.4125f, 0.0f);
glEnd();

//2

glColor3f (0.123, 1.0, 0.983);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.375f, 0.0f);
```

```
glVertex3f(0.275f, 0.5625f, 0.0f);
glVertex3f(0.375f, 0.5375f, 0.0f);
glVertex3f(0.375f, 0.375f, 0.0f);
glEnd();

 glColor3f (0.67, 0.18, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.375f, 0.0f);
glVertex3f(0.275f, 0.55f, 0.0f);
glVertex3f(0.3625f, 0.525f, 0.0f);
glVertex3f(0.3625f, 0.375f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.5f, 0.0f);
glVertex3f(0.275f, 0.525f, 0.0f);
glVertex3f(0.325f, 0.525f, 0.0f);
glVertex3f(0.325f, 0.5f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.4625f, 0.0f);
```

```
glVertex3f(0.275f, 0.4875f, 0.0f);
glVertex3f(0.325f, 0.4875f, 0.0f);
glVertex3f(0.325f, 0.4625f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.425f, 0.0f);
glVertex3f(0.275f, 0.45f, 0.0f);
glVertex3f(0.325f, 0.45f, 0.0f);
glVertex3f(0.325f, 0.425f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.275f, 0.3875f, 0.0f);
glVertex3f(0.275f, 0.4125f, 0.0f);
glVertex3f(0.325f, 0.4125f, 0.0f);
glVertex3f(0.325f, 0.3875f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.337f, 0.5f, 0.0f);
```

```
glVertex3f(0.3375f, 0.525f, 0.0f);
glVertex3f(0.3625f, 0.5125f, 0.0f);
glVertex3f(0.3625f, 0.5f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.3375f, 0.4625f, 0.0f);
glVertex3f(0.3375f, 0.4875f, 0.0f);
glVertex3f(0.3625f, 0.4875f, 0.0f);
glVertex3f(0.3625f, 0.4625f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.3375f, 0.425f, 0.0f);
glVertex3f(0.3375f, 0.45f, 0.0f);
glVertex3f(0.3625f, 0.45f, 0.0f);
glVertex3f(0.3625f, 0.425f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.3375f, 0.3875f, 0.0f);
```

```
glVertex3f(0.3375f, 0.4125f, 0.0f);
glVertex3f(0.3625f, 0.4125f, 0.0f);
glVertex3f(0.3625f, 0.3875f, 0.0f);
glEnd();

//3

 glColor3f (0.67, 0.18, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.375f, 0.375f, 0.0f);
glVertex3f(0.375f, 0.5375f, 0.0f);
glVertex3f(0.5f, 0.5375f, 0.0f);
glVertex3f(0.5f, 0.375f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.3875f, 0.5f, 0.0f);
glVertex3f(0.3875f, 0.525f, 0.0f);
glVertex3f(0.4875f, 0.525f, 0.0f);
glVertex3f(0.4875f, 0.5f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
```

```
 //Trapezoid
 glVertex3f(0.3875f, 0.4625f, 0.0f);
 glVertex3f(0.3875f, 0.4875f, 0.0f);
 glVertex3f(0.4875f, 0.4875f, 0.0f);
 glVertex3f(0.4875f, 0.4625f, 0.0f);
 glEnd();

glColor3f(1.0, 0.9, 0.3);
 glBegin(GL_QUADS);
 //Trapezoid
 glVertex3f(0.3875f, 0.425f, 0.0f);
 glVertex3f(0.3875f, 0.45f, 0.0f);
 glVertex3f(0.4875f, 0.45f, 0.0f);
 glVertex3f(0.4875f, 0.425f, 0.0f);
 glEnd();

 glColor3f(1.0, 0.9, 0.3);
 glBegin(GL_QUADS);
 //Trapezoid
 glVertex3f(0.3875f, 0.3875f, 0.0f);
 glVertex3f(0.3875f, 0.4175f, 0.0f);
 glVertex3f(0.4875f, 0.4175f, 0.0f);
 glVertex3f(0.4875f, 0.3875f, 0.0f);
 glEnd();

 // 4
```

```
 glColor3f (0.67, 0.18, 0.33);
glBegin(GL_POLYGON);
//polygon
glVertex3f(0.5f, 0.375f, 0.0f);
glVertex3f(0.5f, 0.5375f, 0.0f);
glVertex3f(0.5625f, 0.525f, 0.0f);
glVertex3f(0.5625f, 0.4875f, 0.0f);
glVertex3f(0.6f, 0.4875f, 0.0f);
glVertex3f(0.6f, 0.4625f, 0.0f);
glVertex3f(0.625f, 0.4625f, 0.0f);
glVertex3f(0.625f, 0.375f, 0.0f);

glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.5125f, 0.5f, 0.0f);
glVertex3f(0.5125f, 0.525f, 0.0f);
glVertex3f(0.5625f, 0.525f, 0.0f);
glVertex3f(0.5625f, 0.5f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.5125f, 0.4625f, 0.0f);
```

```
glVertex3f(0.5125f, 0.4875f, 0.0f);
glVertex3f(0.6f, 0.4875f, 0.0f);
glVertex3f(0.6f, 0.4625f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.5125f, 0.425f, 0.0f);
glVertex3f(0.5125f, 0.45f, 0.0f);
glVertex3f(0.625f, 0.45f, 0.0f);
glVertex3f(0.625f, 0.425f, 0.0f);
glEnd();

glColor3f(1.0, 0.9, 0.3);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.5125f, 0.3875f, 0.0f);
glVertex3f(0.5125f, 0.4125f, 0.0f);
glVertex3f(0.625f, 0.4125f, 0.0f);
glVertex3f(0.625f, 0.3875f, 0.0f);
glEnd();

//faka jayga ab1

glColor3f (0.67, 0.18, 0.33);
glBegin(GL_QUADS);
```

```
glVertex3f(0.275, 0.55f, 0.0f);
glVertex3f(0.275f, 0.5625f, 0.0f);
glVertex3f(0.375f, 0.5375f, 0.0f);
glVertex3f(0.3625f, 0.525f, 0.0f);



glEnd();

   //faka jayga ab1

 glColor3f (0.67, 0.18, 0.33);
glBegin(GL_QUADS);

glVertex3f(0.375, 0.5375f, 0.0f);
glVertex3f(0.375f, 0.375f, 0.0f);
glVertex3f(0.3625f, 0.375f, 0.0f);
glVertex3f(0.3625f, 0.55f, 0.0f);



glEnd();
```

```
// Mosjid

//1

glColor3f (0.0, 1.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.65f, 0.38f, 0.0f);
glVertex3f(0.84f, 0.38f, 0.0f);
glVertex3f(0.84f, 0.37f, 0.0f);
glVertex3f(0.65f, 0.37f, 0.0f);
glEnd();

//2

glColor3f (1.0, 0.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.61f, 0.53f, 0.0f);
glVertex3f(0.88f, 0.53f, 0.0f);
glVertex3f(0.88f, 0.52f, 0.0f);
glVertex3f(0.61f, 0.52f, 0.0f);
glEnd();

//3
```

```
glColor3f (0.0, 0.0, 1.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.62f, 0.52f, 0.0f);
glVertex3f(0.87f, 0.52f, 0.0f);
glVertex3f(0.83f, 0.38f, 0.0f);
glVertex3f(0.66f, 0.38f, 0.0f);
glEnd();

// 4

glColor3f (1.0, 1.0, 1.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.68f, 0.51f, 0.0f);
glVertex3f(0.8f, 0.51f, 0.0f);
glVertex3f(0.8f, 0.5f, 0.0f);
glVertex3f(0.68f, 0.5f, 0.0f);
glEnd();

//5

glColor3f (1.0, 0.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.68f, 0.48f, 0.0f);
glVertex3f(0.8f, 0.48f, 0.0f);
```

```
glVertex3f(0.8f, 0.38f, 0.0f);
glVertex3f(0.68f, 0.38f, 0.0f);
glEnd();

//6

glColor3f (1.0, 1.0, 1.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.7f, 0.46f, 0.0f);
glVertex3f(0.78f, 0.46f, 0.0f);
glVertex3f(0.78f, 0.38f, 0.0f);
glVertex3f(0.7f, 0.38f, 0.0f);
glEnd();

glColor3f (1.0, 0.0, 0.0);
glBegin(GL_POLYGON);
//Trapezoid
glVertex3f(0.72f, 0.38f, 0.0f);
glVertex3f(0.72f, 0.43f, 0.0f);
glVertex3f(0.74375f, 0.44625f, 0.0f);
glVertex3f(0.76f, 0.43f, 0.0f);
glVertex3f(0.76f, 0.38f, 0.0f);
glEnd();

//extra: line draw
```

```
/*
glBegin(GL_LINES);
glColor3f(0.0,0.0,0.0);
glVertex2f(0,0.3125);
glVertex2f(1, 0.3125);
glEnd();
glFlush();*/

// tree no 1

glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.9f, 0.375f, 0.0f);
glVertex3f(0.925f, 0.375f, 0.0f);
glVertex3f(0.925f, 0.4375f, 0.0f);
glVertex3f(0.9f, 0.4375f, 0.0f);
glEnd();

// leaf 1
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.875f, 0.4375f, 0.0f);
glVertex3f(0.95f, 0.4375f, 0.0f);
glVertex3f(0.9125f, 0.4875f, 0.0f);
glEnd();
```

```
// leaf 2
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.875f, 0.4625f, 0.0f);
glVertex3f(0.95f, 0.4625f, 0.0f);
glVertex3f(0.9125f, 0.5125f, 0.0f);
glEnd();

// leaf 3
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.875f, 0.4875f, 0.0f);
glVertex3f(0.95f, 0.4875f, 0.0f);
glVertex3f(0.9125f, 0.5375f, 0.0f);
glEnd();

// leaf 4
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.875f, 0.5125f, 0.0f);
glVertex3f(0.95f, 0.5125f, 0.0f);
glVertex3f(0.9125f, 0.5625f, 0.0f);
glEnd();
```

```
// main road

glColor3f (0.27, 0.27, 0.47);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0f, 0.375f, 0.0f);
glVertex3f(1.0f, 0.375f, 0.0f);
glVertex3f(1.0f, 0.3125f, 0.0f);
glVertex3f(0.0f, 0.3125f, 0.0f);
glEnd();

// sub road

 glColor3f (0.27, 0.27, 0.47);
 glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.25f, 0.3125f, 0.0f);
glVertex3f(0.3125f, 0.3125f, 0.0f);
glVertex3f(0.2f, 0.0f, 0.0f);
glVertex3f(0.1375f, 0.0f, 0.0f);
glEnd();


// tree no 2

glColor3f (0.50, 0.50, 0.00);
```

```
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.9625f, 0.375f, 0.0f);
glVertex3f(0.975f, 0.375f, 0.0f);
glVertex3f(0.975f, 0.425f, 0.0f);
glVertex3f(0.9625f, 0.425f, 0.0f);
glEnd();

// leaf 1
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.9375f, 0.425f, 0.0f);
glVertex3f(1.0f, 0.425f, 0.0f);
glVertex3f(0.975f, 0.4625f, 0.0f);
glEnd();

// leaf 2
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.9375f, 0.45f, 0.0f);
glVertex3f(1.0f, 0.45f, 0.0f);
glVertex3f(0.975f, 0.4875f, 0.0f);
glEnd();

// leaf 3
```

```
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.9625f, 0.475f, 0.0f);
glVertex3f(0.9875f, 0.475f, 0.0f);
glVertex3f(0.975f, 0.5125f, 0.0f);
glEnd();


// football field

glColor3f (0.20, 0.60, 0.);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.2f, 0.0f, 0.0f);
glVertex3f(0.3125f, 0.3125f, 0.0f);
glVertex3f(1.0f, 0.3125f, 0.0f);
glVertex3f(1.0f, 0.0f, 0.0f);
glEnd();


//dhshd
//Football field
glBegin(GL_LINES);
glColor3f (1.0, 1.0, 1.0);
glVertex2f(0.375,0.25);
glVertex2f(0.9375,0.25);
```

```
    glEnd();

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.9375,0.25);
    glVertex2f(0.8125,0.025);
    glEnd();

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.8125,0.025);
    glVertex2f(0.2875,0.025);
    glEnd();

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.375,0.25);
    glVertex2f(0.2875,0.025);
    glEnd();

//Football field bar left

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.3625,0.225);
    glVertex2f(0.475,0.225);
    glEnd();
```

```
    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.475,0.225);
    glVertex2f(0.4,0.05);
    glEnd();

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.3,0.05);
    glVertex2f(0.4,0.05);
    glEnd();

// middle line of football field

    glBegin(GL_LINES);
    glColor3f (1.0, 1.0, 1.0);
    glVertex2f(0.6625,0.25);
    glVertex2f(0.55,0.025);
    glEnd();

// football right bar

    glBegin(GL_LINES);
    glColor3f (1, 1.0, 1.0);
    glVertex2f(0.8125,0.225);
    glVertex2f(0.925,0.225);
```

```
glEnd();

glBegin(GL_LINES);
glColor3f (1.0, 1.0, 1.0);
glVertex2f(0.8125,0.225);
glVertex2f(0.7125,0.05);
glEnd();

glBegin(GL_LINES);
glColor3f (1.0, 1.0, 1.0);
glVertex2f(0.7125,0.05);
glVertex2f(0.825,0.05);
glEnd();
// innovation lab

glColor3f (1.0, 0.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0f, 0.5f, 0.0f);
glVertex3f(0.0f, 0.55f, 0.0f);
glVertex3f(0.1875f, 0.55f, 0.0f);
glVertex3f(0.1875f, 0.5f, 0.0f);
glEnd();

glColor3f (1.0, 1.0, 1.0);
glBegin(GL_QUADS);
//Trapezoid
```

```
glVertex3f(0.025f, 0.5f, 0.0f);
glVertex3f(0.1625f, 0.5f, 0.0f);
glVertex3f(0.1625f, 0.4375f, 0.0f);
glVertex3f(0.025f, 0.4375f, 0.0f);
glEnd();

// small box 1

glColor3f (0.33, 0.33, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0f, 0.55f, 0.0f);
glVertex3f(0.0375f, 0.55f, 0.0f);
glVertex3f(0.0375f, 0.525f, 0.0f);
glVertex3f(0.0f, 0.525f, 0.0f);
glEnd();

// 2

glColor3f (0.0, 1.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0375f, 0.55f, 0.0f);
glVertex3f(0.075f, 0.55f, 0.0f);
glVertex3f(0.075f, 0.55f, 0.0f);
glVertex3f(0.0375f, 0.55f, 0.0f);
glEnd();
```

```
// 3

glColor3f (0.33, 0.33, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.075f, 0.55f, 0.0f);
glVertex3f(0.1125f, 0.55f, 0.0f);
glVertex3f(0.1125f, 0.525f, 0.0f);
glVertex3f(0.075f, 0.525f, 0.0f);
glEnd();

// 4

glColor3f (0.0, 1.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.1125f, 0.55f, 0.0f);
glVertex3f(0.15f, 0.55f, 0.0f);
glVertex3f(0.15f, 0.525f, 0.0f);
glVertex3f(0.1125f, 0.525f, 0.0f);
glEnd();

// 5

glColor3f (0.33, 0.33, 0.33);
glBegin(GL_QUADS);
```

```
//Trapezoid
glVertex3f(0.15f, 0.55f, 0.0f);
glVertex3f(0.1875f, 0.55f, 0.0f);
glVertex3f(0.1875f, 0.525f, 0.0f);
glVertex3f(0.15f, 0.525f, 0.0f);
glEnd();

// 6

glColor3f (0.0, 1.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0f, 0.525f, 0.0f);
glVertex3f(0.0375f, 0.525f, 0.0f);
glVertex3f(0.0375f, 0.5f, 0.0f);
glVertex3f(0.0f, 0.5f, 0.0f);
glEnd();

// 7

glColor3f (0.33, 0.33, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0375f, 0.525f, 0.0f);
glVertex3f(0.075f, 0.525f, 0.0f);
glVertex3f(0.075f, 0.5f, 0.0f);
glVertex3f(0.0375f, 0.5f, 0.0f);
```

```
glEnd();

// 8

glColor3f (0.0, 1.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.075f, 0.525f, 0.0f);
glVertex3f(0.1125f, 0.525f, 0.0f);
glVertex3f(0.1125f, 0.5f, 0.0f);
glVertex3f(0.075f, 0.5f, 0.0f);
glEnd();

// 9

glColor3f (0.33, 0.33, 0.33);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.1125f, 0.525f, 0.0f);
glVertex3f(0.15f, 0.525f, 0.0f);
glVertex3f(0.15f, 0.5f, 0.0f);
glVertex3f(0.1125f, 0.5f, 0.0f);
glEnd();

// 10

glColor3f (0.0, 1.0, 0.0);
```

```
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.15f, 0.525f, 0.0f);
glVertex3f(0.1875f, 0.525f, 0.0f);
glVertex3f(0.1875f, 0.5f, 0.0f);
glVertex3f(0.15f, 0.5f, 0.0f);
glEnd();

// lower side door

glColor3f (1.0, 0.0, 0.0);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0125f, 0.375f, 0.0f);
glVertex3f(0.175f, 0.375f, 0.0f);
glVertex3f(0.175f, 0.4375f, 0.0f);
glVertex3f(0.0125f, 0.4375f, 0.0f);
glEnd();

// door

glColor3f (0.0, 0.33, 0.67);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.075f, 0.375f, 0.0f);
glVertex3f(0.1125f, 0.375f, 0.0f);
glVertex3f(0.1125f, 0.4625f, 0.0f);
```

```
glVertex3f(0.075f, 0.4625f, 0.0f);
glEnd();

// small field

glColor3f (0.20, 0.60, 0.);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0f, 0.0f, 0.0f);
glVertex3f(0.1375f, 0.0f, 0.0f);
glVertex3f(0.25f, 0.3125f, 0.0f);
glVertex3f(0.0f, 0.3125f, 0.0f);
glEnd();

// small field tree 1
 glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.0625f, 0.0625f, 0.0f);
glVertex3f(0.0875f, 0.0625f, 0.0f);
glVertex3f(0.0875f, 0.0f, 0.0f);
glVertex3f(0.0625f, 0.0f, 0.0f);
glEnd();

// leaf1
glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
```

```
 //Trapezoid
 glVertex3f(0.075f, 0.1875f, 0.0f);
 glVertex3f(0.1125f, 0.125f, 0.0f);
 glVertex3f(0.0375f, 0.125f, 0.0f);
 glEnd();

 // leaf2
glColor3f (0.25, 1, 0.0);
 glBegin(GL_TRIANGLES);
 //Trapezoid
 glVertex3f(0.075f, 0.15f, 0.0f);
 glVertex3f(0.1125f, 0.0875f, 0.0f);
 glVertex3f(0.0375f, 0.0875f, 0.0f);
 glEnd();

 // leaf3
glColor3f (0.25, 1, 0.0);
 glBegin(GL_TRIANGLES);
 //Trapezoid
 glVertex3f(0.075f, 0.1125f, 0.0f);
 glVertex3f(0.1125f, 0.0625f, 0.0f);
 glVertex3f(0.0375f, 0.0625f, 0.0f);
 glEnd();


 ///Bus
 bus();
```

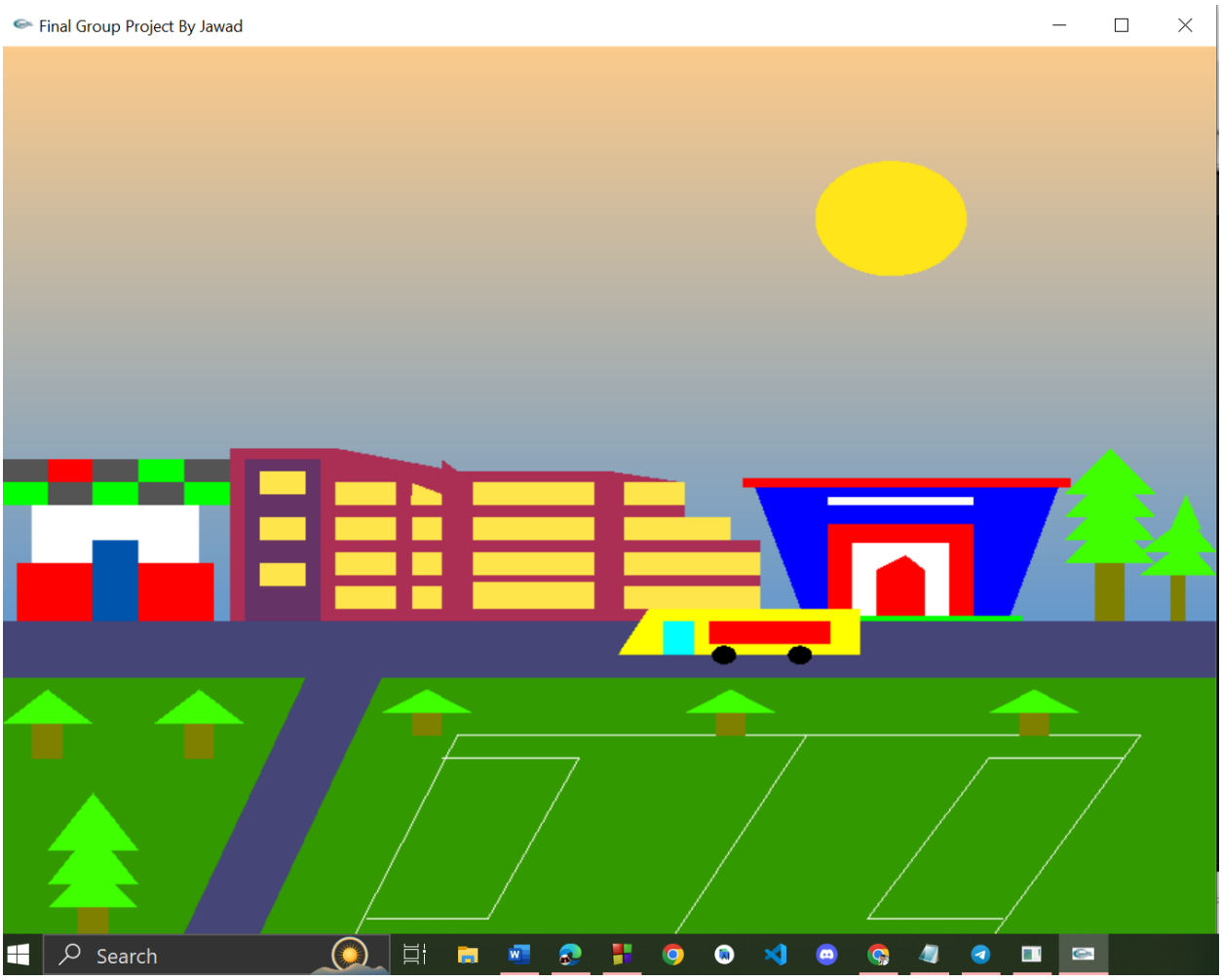
```
// field side tree 1
   glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.0f, 0.2625f, 0.0f);
glVertex3f(0.075f, 0.2625f, 0.0f);
glVertex3f(0.0375f, 0.3f, 0.0f);
glEnd();
    glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.025f, 0.225f, 0.0f);
glVertex3f(0.05f, 0.225f, 0.0f);
glVertex3f(0.05f, 0.2625f, 0.0f);
glVertex3f(0.025f, 0.2625f, 0.0f);
glEnd();

// field side tree 2
    glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.125f, 0.2625f, 0.0f);
glVertex3f(0.2f, 0.2625f, 0.0f);
glVertex3f(0.1625f, 0.3f, 0.0f);
```

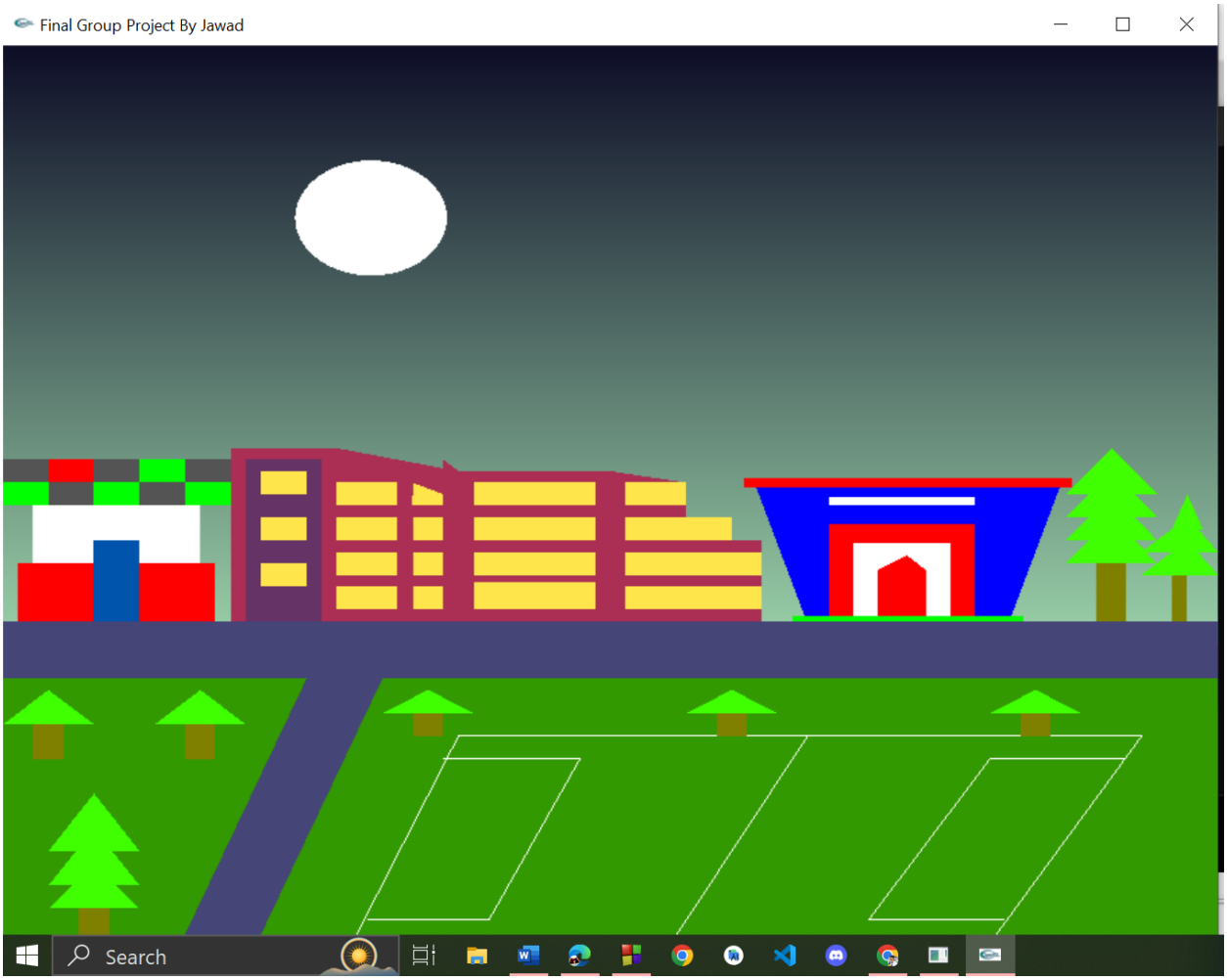
```
glEnd();
   glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.15f, 0.225f, 0.0f);
glVertex3f(0.175f, 0.225f, 0.0f);
glVertex3f(0.175f, 0.2625f, 0.0f);
glVertex3f(0.15f, 0.2625f, 0.0f);
glEnd();
// field side tree 3
  glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.3125f, 0.275f, 0.0f);
glVertex3f(0.3875f, 0.275f, 0.0f);
glVertex3f(0.35f, 0.3f, 0.0f);
glEnd();
   glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.3375f, 0.25f, 0.0f);
glVertex3f(0.3625f, 0.25f, 0.0f);
glVertex3f(0.3625f, 0.275f, 0.0f);
glVertex3f(0.3375f, 0.275f, 0.0f);
glEnd();
// field side tree 4
  glColor3f (0.25, 1, 0.0);
```

```
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.5625f, 0.275f, 0.0f);
glVertex3f(0.6375f, 0.275f, 0.0f);
glVertex3f(0.6f, 0.3f, 0.0f);
glEnd();
   glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.5875f, 0.25f, 0.0f);
glVertex3f(0.6125f, 0.25f, 0.0f);
glVertex3f(0.6125f, 0.275f, 0.0f);
glVertex3f(0.5875f, 0.275f, 0.0f);
glEnd();
// field side tree 5
   glColor3f (0.25, 1, 0.0);
glBegin(GL_TRIANGLES);
//Trapezoid
glVertex3f(0.8125f, 0.275f, 0.0f);
glVertex3f(0.8875f, 0.275f, 0.0f);
glVertex3f(0.85f, 0.3f, 0.0f);
glEnd();
  glColor3f (0.50, 0.50, 0.00);
glBegin(GL_QUADS);
//Trapezoid
glVertex3f(0.8375f, 0.25f, 0.0f);
glVertex3f(0.8625f, 0.25f, 0.0f);
```

```
    glVertex3f(0.8625f, 0.275f, 0.0f);
    glVertex3f(0.8375f, 0.275f, 0.0f);
    glEnd();



    glutSwapBuffers();
}

int main(int argc,char **argv)
{



    glutInit(&argc,argv);
    glutInitDisplayMode ( GLUT_RGB | GLUT_DOUBLE );
    glutInitWindowPosition(0,0);
    glutInitWindowSize(900,800);
    glutCreateWindow("Final Group Project By Jawad");
    init();
    glutDisplayFunc(Draw);
    glutTimerFunc(0, update, 0);
    glutMainLoop();
    return 0;
}
```

Output:


Final Group Project By Jawad
Search


Final Group Project By Jawad
Search