Computer Graphis Project Report

Team Member:

Id: 201-15-3293 (D1)
Id: 201-15-3187 (D1)
Id: 201-15-3191 (D2)
Id: 201-15-3176 (D2)

Title: Smart Village

Code:

```
#include <windows.h>
#define PI 3.14159265358979323846
#include<math.h>
#include <GL/gl.h>
#include <GL/glut.h>



float xPos = 0.0f;
float yPos = 0.0f;
float x1Pos = 0.0f;
float xP = 0.0f;
float p=0,q=0,f=0;
float u=0,v=0,w=0;
float angle = 0.0f;

float translateX = 0.0f;
float translateY = 0.0f;


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

}
void display()
```

```
{
   glClear(GL_COLOR_BUFFER_BIT);
   glLoadIdentity();



//background.....

   glBegin(GL_QUADS);
   glColor3ub (p,q,f);
//glColor3ub(51,204,255);
   glVertex2f(0.0,0.675);
   glVertex2f(1.0,0.675);
   glColor3ub(0,102,204);
   glVertex2f(1.0,1.0);
   glVertex2f(0.0,1.0);
   glEnd();



   glPushMatrix();
   glLoadIdentity();
   glTranslatef(translateX, translateY, 0.0f);

//sun.....

   int i;
   GLfloat x = .5f;
   GLfloat y = .9f;
   GLfloat radius = .06f;
   int triangleAmount = 20;
   GLfloat twicePi = 2.0f * PI;
   glBegin(GL_TRIANGLE_FAN);

   glColor3f (u,v,w);
   glVertex2f(x, y); // center of circle
   for (i = 0; i <= triangleAmount; i++)
   {
      glVertex2f(
         x + (radius * cos(i * twicePi / triangleAmount)),
         y + (radius * sin(i * twicePi / triangleAmount))
      );
   }
   glEnd();

   glPopMatrix();
```

```
glPushMatrix();
glLoadIdentity();
glTranslatef(0.0f, yPos, 0.0f);
//star.....................................................................

glBegin(GL_QUADS);
glColor3f (0.70, 0.70, 0.0);
glVertex3f(0.90f, 0.85f, 0.0f);
glVertex3f(0.92f, 0.85f, 0.0f);
glVertex3f(0.92f, 0.88f, 0.0f);
glVertex3f(0.90f, 0.88f, 0.0f);

glVertex3f(0.34f, 0.88f, 0.0f);
glVertex3f(0.36f, 0.88f, 0.0f);
glVertex3f(0.36f, 0.91f, 0.0f);
glVertex3f(0.34f, 0.91f, 0.0f);
glEnd();


glBegin(GL_TRIANGLES);

glColor3f (0.70, 0.70, 0.0);
glVertex3f(0.90f, 0.85f, 0.0f);
glVertex3f(0.92f, 0.85f, 0.0f);
glVertex3f(0.91f, 0.80f, 0.0f);

glVertex3f(0.92f, 0.85f, 0.0f);
glVertex3f(0.92f, 0.88f, 0.0f);
glVertex3f(0.95f, 0.86f, 0.0f);

glVertex3f(0.92f, 0.88f, 0.0f);
glVertex3f(0.90f, 0.88f, 0.0f);
glVertex3f(0.91f, 0.93f, 0.0f);

glVertex3f(0.90f, 0.88f, 0.0f);
glVertex3f(0.90f, 0.85f, 0.0f);
glVertex3f(0.87f, 0.86f, 0.0f);

glVertex3f(0.34f, 0.88f, 0.0f);
glVertex3f(0.36f, 0.88f, 0.0f);
glVertex3f(0.35f, 0.83f, 0.0f);

glVertex3f(0.36f, 0.88f, 0.0f);
glVertex3f(0.36f, 0.91f, 0.0f);
glVertex3f(0.39f, 0.90f, 0.0f);
```

```
    glVertex3f(0.36f, 0.91f, 0.0f);
    glVertex3f(0.34f, 0.91f, 0.0f);
    glVertex3f(0.35f, 0.96f, 0.0f);

    glVertex3f(0.34f, 0.91f, 0.0f);
    glVertex3f(0.34f, 0.88f, 0.0f);
    glVertex3f(0.31f, 0.90f, 0.0f);

    glEnd();
    glPopMatrix();



//river.....................................................................................

    glBegin(GL_QUADS);

    glColor3ub(34,139,34);
    glVertex2f(0.0,0.0);
    glColor3ub(102,178,255);
    glVertex2f(1.0,0.0);

    glColor3ub(0,128,255);
    glVertex2f(1,0.675);
    glColor3ub(1,76,153);
    glVertex2f(0.0,0.675);

//Green shape groud.....................................................................................
    glBegin(GL_POLYGON);
    glColor3ub(34,139,34);
    glVertex2f(0.0,0.0);
    glVertex2f(0.67,0);
    glVertex2f(0.63,0.025);
    glVertex2f(0.0,0.025);

    glVertex2f(0.63,0.025);
    glVertex2f(0.67,0.05);
    glVertex2f(0.61,0.075);
    glVertex2f(0.0,0.075);

    glVertex2f(0.61,0.075);
    glVertex2f(0.65,0.0875);
    glVertex2f(0.6,0.1125);
    glVertex2f(0.0,0.1125);

    glVertex2f(0.6,0.1125);
    glVertex2f(0.66,0.1125);
```

```
    glVertex2f(0.69,0.1875);
    glVertex2f(0.6,0.1875);
    glVertex2f(0.0,0.1875);

    glVertex2f(0.6,0.1875);
    glVertex2f(0.64,0.2125);
    glVertex2f(0.6,0.225);
    glVertex2f(0.0,0.225);
glVertex2f(0.6,0.225);
    glVertex2f(0.67,0.25);
    glVertex2f(0.58,0.2875);
    glVertex2f(0.0,0.2875);

    glVertex2f(0.58,0.2875);
    glVertex2f(0.65,0.325);
    glVertex2f(0.5,0.4125);
    glVertex2f(0.0,0.4125);
    glEnd();



//tree.....

    glBegin(GL_QUADS);
    glColor3f(0.43,0.3,0.08);
    glVertex2f(0.372,0.675);
    glVertex2f(0.378,0.675);
    glVertex2f(0.378,0.7);
    glVertex2f(0.372,0.7);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0.07,0.5,0.07);
    glVertex2f(0.36,0.7);
    glVertex2f(0.39,0.7);
    glVertex2f(0.375,0.75);
    glEnd();

    glBegin(GL_TRIANGLES);
    glVertex2f(0.36,0.725);
    glVertex2f(0.39,0.725);
    glVertex2f(0.375,0.75);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0,1,0);
    glColor3f(0.07,0.5,0.07);
```

```
    glVertex2f(0.36,0.7125);
    glVertex2f(0.39,0.7125);
    glVertex2f(0.375,0.75);
    glEnd();

//small tree.....
    glBegin(GL_QUADS);
    glColor3f(0.43,0.3,0.08);
    glVertex2f(0.242,0.675);
    glVertex2f(0.248,0.675);
    glVertex2f(0.248,0.7125);
    glVertex2f(0.242,0.7125);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0.07,0.5,0.07);
    glVertex2f(0.23,0.7125);
    glVertex2f(0.26,0.7125);
    glVertex2f(0.245,0.7625);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0.07,0.5,0.07);
    glVertex2f(0.23,0.7);
    glVertex2f(0.26,0.7);
    glVertex2f(0.245,0.7625);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0.07,0.5,0.07);
    glVertex2f(0.23,0.725);
    glVertex2f(0.26,0.725);
    glVertex2f(0.245,0.7625);
    glEnd();




//hill.....

    glBegin(GL_POLYGON);
    glColor3ub(155, 118, 83);
    glVertex2f(0.0,0.675);
    glVertex2f(0.25,0.675);
    glVertex2f(0.18,0.7);
    glVertex2f(0.1,0.75);
    glEnd();
```

```
    glBegin(GL_POLYGON);
    glColor3ub(155, 118, 83);
    glVertex2f(0.25,0.675);
    glVertex2f(0.37,0.675);
    glVertex2f(0.35,0.725);
    glVertex2f(0.33,0.7125);
    glVertex2f(0.3,0.725);
    glEnd();

    glBegin(GL_POLYGON);
    glColor3ub(155, 118, 83);
    glVertex2f(0.37,0.675);
    glVertex2f(0.65,0.675);
    glVertex2f(0.59,0.7125);
    glVertex2f(0.57,0.7125);
    glVertex2f(0.5,0.75);
    glEnd();

    glBegin(GL_POLYGON);
    glColor3ub(155, 118, 83);
    glVertex2f(0.65,0.675);
    glVertex2f(1,0.675);
    glVertex2f(0.89,0.75);
    glVertex2f(0.82,0.6875);
    glVertex2f(0.77,0.6875);
    glVertex2f(0.73,0.725);
    glVertex2f(0.7,0.7);
    glVertex2f(0.68,0.7125);
    glEnd();



//1st stand Boat .....


    glBegin(GL_QUADS);
    glColor3f(0,0,0);
    glVertex2f(0.72,0.1625);
    glVertex2f(0.88,0.1625);
    glVertex2f(0.9,0.1875);
    glVertex2f(0.7,0.1875);
    glEnd();

    glBegin(GL_POLYGON);
    glColor3ub(153, 76, 0);
    glVertex2f(0.74,0.1875);
    glVertex2f(0.86,0.1875);
```

```
glVertex2f(0.86,0.2375);
glVertex2f(0.85,0.25);
glVertex2f(0.75,0.25);
glVertex2f(0.74,0.2375);
glEnd();


glBegin(GL_QUADS);
glColor3ub(0, 0, 204);
glVertex2f(0.75,0.265);
glVertex2f(0.83,0.2525);
glVertex2f(0.83,0.35);
glVertex2f(0.75,0.3625);
glEnd();

glBegin(GL_QUADS);
glColor3f(0,0,0);
glVertex2f(0.79,0.25);
glVertex2f(0.8,0.25);
glVertex2f(0.8,0.37);
glVertex2f(0.79,0.37);
glEnd();



 glPushMatrix();
glLoadIdentity();
glTranslatef(xPos, 0.0, 0.0f);
//3rd boat.....

glBegin(GL_QUADS);
glColor3ub(102, 0, 1);
glVertex2f(0.57,0.5625);
glVertex2f(0.73,0.5625);
glVertex2f(0.75,0.5875);
glVertex2f(0.55,0.5875);
glEnd();

glBegin(GL_POLYGON);
glColor3ub(0, 0, 102);
glVertex2f(0.59,0.5875);
glVertex2f(0.7,0.5875);
glVertex2f(0.7,0.6375);
glVertex2f(0.69,0.65);
glVertex2f(0.6,0.65);
glVertex2f(0.59,0.6375);
glEnd();
```

```
    glBegin(GL_QUADS);
    glColor3f(1,0,0);
    glVertex2f(0.61,0.665);
    glVertex2f(0.67,0.6525);
    glVertex2f(0.67,0.75);
    glVertex2f(0.61,0.76);
    glEnd();


    glBegin(GL_QUADS);
    glColor3f(0,0,0);
    glVertex2f(0.64,0.65);
    glVertex2f(0.65,0.65);
    glVertex2f(0.65,0.7725);
    glVertex2f(0.64,0.7725);
    glEnd();


    glLoadIdentity();

    glPopMatrix();


    glPushMatrix();
    glLoadIdentity();
    glTranslatef(xP, 0.0, 0.0f);

    //2nd boat.....................................................................................

    glBegin(GL_QUADS);
    glColor3ub(102, 51, 0);
    glVertex2f(0.77,0.4375);
    glVertex2f(0.93,0.4375);
    glVertex2f(0.95,0.4625);
    glVertex2f(0.75,0.4625);
    glEnd();
  glBegin(GL_POLYGON);
    glColor3f(0.69,0.75,0.22);
    glVertex2f(0.8,0.4625);
    glVertex2f(0.91,0.4625);
    glVertex2f(0.91,0.5125);
    glVertex2f(0.9,0.525);
    glVertex2f(0.81,0.525);
    glVertex2f(0.8,0.5125);
    glEnd();

    glBegin(GL_QUADS);
```

```
    glColor3f(0.88,0.88,0.88);
    glVertex2f(0.81,0.55);
    glVertex2f(0.89,0.525);
    glVertex2f(0.89,0.63);
    glVertex2f(0.81,0.64);
    glEnd();


    glBegin(GL_QUADS);
    glColor3f(0,0,0);
    glVertex2f(0.85,0.525);
    glVertex2f(0.86,0.525);
    glVertex2f(0.86,0.65);
    glVertex2f(0.85,0.65);
    glEnd();
    glLoadIdentity();
    glPopMatrix();




//river carl.....

    glBegin(GL_QUADS);
    glColor3f(0.62,0.48,0.2);
    glVertex2f(0.6,0.28125);
    glVertex2f(0.65,0.3125);
    glVertex2f(0.65,0.325);
    glVertex2f(0.58,0.2875);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(0.62,0.48,0.2);
    glVertex2f(0.62,0.21875);
    glVertex2f(0.67,0.2375);
    glVertex2f(0.67,0.25);
    glVertex2f(0.6,0.225);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(0.62,0.48,0.2);
    glVertex2f(0.61,0.1875);
    glVertex2f(0.64,0.2);
    glVertex2f(0.64,0.2125);
    glVertex2f(0.6,0.1875);
    glEnd();
```

```
glBegin(GL_QUADS);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.63,0.1);
glVertex2f(0.67,0.1);
glVertex2f(0.66,0.1125);
glVertex2f(0.6,0.1125);
glEnd();

glBegin(GL_TRIANGLES);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.67,0.1);
glVertex2f(0.69,0.1875);
glVertex2f(0.66,0.1125);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.63,0.06875);
glVertex2f(0.65,0.075);
glVertex2f(0.65,0.0875);
glVertex2f(0.61,0.075);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.64,0.01875);
glVertex2f(0.67,0.0375);
glVertex2f(0.67,0.05);
glVertex2f(0.63,0.025);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.67,0.175);
glVertex2f(0.68,0.175);
glVertex2f(0.68,0.225);
glVertex2f(0.67,0.225);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.62,0.48,0.2);
glVertex2f(0.68,0.2);
glVertex2f(0.71,0.1875);
glVertex2f(0.72,0.1875);
glVertex2f(0.68,0.2125);
glEnd();
```

```
//House drawing.....................................................................................

   glBegin(GL_QUADS);
   glColor3f(0.96,0.94,0.87);
   glVertex2f(0.25,0.125);
   glVertex2f(0.4,0.125);
   glVertex2f(0.4,0.2);
   glVertex2f(0.25,0.2);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(1,0,0);
   glVertex2f(0.25,0.2);
   glVertex2f(0.4,0.2);
   glVertex2f(0.4,0.2125);
   glVertex2f(0.25,0.2125);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(0.96,0.94,0.87);
   glVertex2f(0.25,0.2125);
   glVertex2f(0.4,0.2125);
   glVertex2f(0.4,0.3);
   glVertex2f(0.25,0.3);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(1,0,0);
   glVertex2f(0.25,0.3);
   glVertex2f(0.4,0.3);
   glVertex2f(0.4,0.3125);
   glVertex2f(0.25,0.3125);
   glEnd();
glBegin(GL_QUADS);
   glColor3f(0.96,0.94,0.87);
   glVertex2f(0.25,0.3125);
   glVertex2f(0.4,0.3125);
   glVertex2f(0.4,0.3875);
   glVertex2f(0.25,0.3875);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(1,0,0);
   glVertex2f(0.25,0.3875);
   glVertex2f(0.4,0.3875);
   glVertex2f(0.4,0.4);
```

```
    glVertex2f(0.25,0.4);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(0.96,0.94,0.87);
    glVertex2f(0.25,0.4);
    glVertex2f(0.4,0.4);
    glVertex2f(0.4,0.475);
    glVertex2f(0.25,0.475);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,0,0);
    glVertex2f(0.25,0.475);
    glVertex2f(0.4,0.475);
    glVertex2f(0.4,0.4875);
    glVertex2f(0.25,0.4875);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(0.96,0.94,0.87);
    glVertex2f(0.25,0.4875);
    glVertex2f(0.4,0.4875);
    glVertex2f(0.4,0.5625);
    glVertex2f(0.25,0.5625);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,0,0);
    glVertex2f(0.25,0.5625);
    glVertex2f(0.4,0.5625);
    glVertex2f(0.4,0.575);
    glVertex2f(0.25,0.575);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,0,0);
    glVertex2f(0.25,0.3);
    glVertex2f(0.4,0.3);
    glVertex2f(0.4,0.3125);
    glVertex2f(0.25,0.3125);
    glEnd();


//middle windows.....

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
```

```
    glVertex2f(0.3,0.125);
    glVertex2f(0.35,0.125);
    glVertex2f(0.35,0.175);
    glVertex2f(0.3,0.175);
    glEnd();

    glBegin(GL_QUADS);
    glColor3ub(96, 96, 96);
    glVertex2f(0.3,0.2375);
    glVertex2f(0.35,0.2375);
    glVertex2f(0.35,0.275);
    glVertex2f(0.3,0.275);
    glEnd();

    glBegin(GL_QUADS);
    glColor3ub(96, 96, 96);
    glVertex2f(0.3,0.325);
    glVertex2f(0.35,0.325);
    glVertex2f(0.35,0.3625);
    glVertex2f(0.3,0.3625);
    glEnd();

    glBegin(GL_QUADS);
    glColor3ub(96, 96, 96);
    glVertex2f(0.3,0.4125);
    glVertex2f(0.35,0.4125);
    glVertex2f(0.35,0.45);
    glVertex2f(0.3,0.45);
    glEnd();

    glBegin(GL_QUADS);
    glColor3ub(96, 96, 96);
    glVertex2f(0.3,0.5);
    glVertex2f(0.35,0.5);
    glVertex2f(0.35,0.5375);
    glVertex2f(0.3,0.5375);
    glEnd();


//  left and right windows.....................................................................

    glBegin(GL_QUADS);
    glColor3f(0.17,0.14,0.14);
    glVertex2f(0.26,0.125);
    glVertex2f(0.28,0.125);
    glVertex2f(0.28,0.175);
    glVertex2f(0.26,0.175);
    glEnd();
```

```
glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.37,0.125);
glVertex2f(0.39,0.125);
glVertex2f(0.39,0.175);
glVertex2f(0.37,0.175);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.26,0.2375);
glVertex2f(0.28,0.2375);
glVertex2f(0.28,0.275);
glVertex2f(0.26,0.275);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.37,0.2375);
glVertex2f(0.39,0.2375);
glVertex2f(0.39,0.275);
glVertex2f(0.37,0.275);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.26,0.325);
glVertex2f(0.28,0.325);
glVertex2f(0.28,0.3625);
glVertex2f(0.26,0.3625);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.37,0.325);
glVertex2f(0.39,0.325);
glVertex2f(0.39,0.3625);
glVertex2f(0.37,0.3625);
glEnd();

glBegin(GL_QUADS);
glColor3f(0.17,0.14,0.14);
glVertex2f(0.26,0.4125);
glVertex2f(0.28,0.4125);
glVertex2f(0.28,0.45);
glVertex2f(0.26,0.45);
glEnd();
```

```
    glBegin(GL_QUADS);
    glColor3f(0.17,0.14,0.14);
    glVertex2f(0.37,0.4125);
    glVertex2f(0.39,0.4125);
    glVertex2f(0.39,0.45);
    glVertex2f(0.37,0.45);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(0.17,0.14,0.14);
    glVertex2f(0.26,0.5);
    glVertex2f(0.28,0.5);
    glVertex2f(0.28,0.5375);
    glVertex2f(0.26,0.5375);
    glEnd();


    glBegin(GL_QUADS);
    glColor3f(0.17,0.14,0.14);
    glVertex2f(0.37,0.5);
    glVertex2f(0.39,0.5);
    glVertex2f(0.39,0.5375);
    glVertex2f(0.37,0.5375);
    glEnd();
  glPushMatrix();
    glLoadIdentity();
    glRotatef(angle, 1.0f, 0.0f, 0.0f);

    //2nd top left.....................................
    glBegin(GL_QUADS);
    glColor3ub(34,139,34);
    glVertex2f(0.265,0.4225);
    glVertex2f(0.275,0.4225);
    glVertex2f(0.275,0.44);
    glVertex2f(0.265,0.44);
    glEnd();

    //2nd top right.....................................
    glBegin(GL_QUADS);
    glColor3ub(34,139,34);
    glVertex2f(0.375,0.4225);
    glVertex2f(0.385,0.4225);
    glVertex2f(0.385,0.44);
    glVertex2f(0.375,0.44);
    glEnd();

    //small left top.....................................
```

```
    glBegin(GL_QUADS);
    glColor3ub(34,139,34);
    glVertex2f(0.265,0.51);
    glVertex2f(0.275,0.51);
    glVertex2f(0.275,0.53);
    glVertex2f(0.265,0.53);
    glEnd();

    //small top right.....
    glBegin(GL_QUADS);
    glColor3ub(34,139,34);
    glVertex2f(0.375,0.51);
    glVertex2f(0.385,0.51);
    glVertex2f(0.385,0.53);
    glVertex2f(0.375,0.53);
    glEnd();


    glPopMatrix();



    //antena.....
    glBegin(GL_QUADS);
    glColor3f(0.96,0.94,0.87);
    glVertex2f(0.26,0.575);
    glVertex2f(0.28,0.575);
    glVertex2f(0.28,0.6);
    glVertex2f(0.26,0.6);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,0,0);
    glVertex2f(0.26,0.6);
    glVertex2f(0.28,0.6);
    glVertex2f(0.28,0.6125);
    glVertex2f(0.26,0.6125);
    glEnd();

    glBegin(GL_TRIANGLES);
    glColor3f(0,0,0);
    glVertex2f(0.26,0.6125);
    glVertex2f(0.28,0.6125);
    glVertex2f(0.27,0.625);
    glEnd();


//road.....
```

```
   glBegin(GL_QUADS);
   glColor3f(0.40,0.40,0.40);
   glVertex2f(0.25,0.0);
   glVertex2f(0.30,0.0);
   glVertex2f(0.35,0.125);
   glVertex2f(0.30,0.125);
   glEnd();

//rode indicator.....
   glBegin(GL_LINES);

   glColor3f (1.0, 1.0, 1.0);
   glVertex2f(0.275f, 0.0f);
   glVertex2f(0.282f, 0.02f);

   glVertex2f(0.29f, 0.04f);
   glVertex2f(0.305f, 0.08f);

   glVertex2f(0.315f, 0.105f);
   glVertex2f(0.325f, 0.125f);


   glEnd();

//Fencing.....
//column.....
   glBegin(GL_QUADS);
   glColor3f(1,1,0);
   glVertex2f(0.01,0.2875);
   glVertex2f(0.02,0.2875);
   glVertex2f(0.02,0.5125);
   glVertex2f(0.01,0.5125);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(1,1,0);
   glVertex2f(0.04,0.2875);
   glVertex2f(0.05,0.2875);
   glVertex2f(0.05,0.5125);
   glVertex2f(0.04,0.5125);
   glEnd();

   glBegin(GL_QUADS);
   glColor3f(1,1,0);
   glVertex2f(0.07,0.2875);
   glVertex2f(0.08,0.2875);
   glVertex2f(0.08,0.5125);
```

```
    glVertex2f(0.07,0.5125);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.1,0.2875);
    glVertex2f(0.11,0.2875);
    glVertex2f(0.11,0.5125);
    glVertex2f(0.1,0.5125);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.13,0.2875);
    glVertex2f(0.14,0.2875);
    glVertex2f(0.14,0.5125);
    glVertex2f(0.13,0.5125);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.16,0.2875);
    glVertex2f(0.17,0.2875);
    glVertex2f(0.17,0.5125);
    glVertex2f(0.16,0.5125);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.19,0.2875);
    glVertex2f(0.2,0.2875);
    glVertex2f(0.2,0.5125);
    glVertex2f(0.19,0.5125);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.22,0.2875);
    glVertex2f(0.23,0.2875);
    glVertex2f(0.23,0.5125);
    glVertex2f(0.22,0.5125);
    glEnd();

//row.....................................
    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.3);
    glVertex2f(0.25,0.3);
```

```
    glVertex2f(0.25,0.3125);
    glVertex2f(0.0,0.3125);
    glEnd();
glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.3375);
    glVertex2f(0.25,0.3375);
    glVertex2f(0.25,0.35);
    glVertex2f(0.0,0.35);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.375);
    glVertex2f(0.25,0.375);
    glVertex2f(0.25,0.3875);
    glVertex2f(0.0,0.3875);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.4125);
    glVertex2f(0.25,0.4125);
    glVertex2f(0.25,0.425);
    glVertex2f(0.0,0.425);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.45);
    glVertex2f(0.25,0.45);
    glVertex2f(0.25,0.4625);
    glVertex2f(0.0,0.4625);
    glEnd();

    glBegin(GL_QUADS);
    glColor3f(1,1,0);
    glVertex2f(0.0,0.375);
    glVertex2f(0.25,0.375);
    glVertex2f(0.25,0.3875);
    glVertex2f(0.0,0.3875);
    glEnd();



    glPushMatrix();
    glLoadIdentity();
    glTranslatef(x1Pos, 0.0, 0.0f);
```

```
//cloud.....................................................................


int j;
GLfloat m = .20f;
GLfloat n = .88f;
GLfloat r = .05f;
int triangleAmount2 = 20;
GLfloat twicePi2 = 2.0f * PI;
glBegin(GL_TRIANGLE_FAN);
glColor3ub(208, 222, 236);
glVertex2f(m, n); // center of circle
for (j = 0; j <= triangleAmount2; j++)
{
   glVertex2f(
      m + (r * cos(j * twicePi2 / triangleAmount2)),
      n + (r * sin(j * twicePi2 / triangleAmount2))
   );
}
glEnd();


GLfloat a = .25f;
GLfloat b = .85f;
glBegin(GL_TRIANGLE_FAN);
glColor3ub(208, 222, 236);
glVertex2f(a, b); // center of circle
for (j = 0; j <= triangleAmount2; j++)
{
   glVertex2f(
      a + (r * cos(j * twicePi2 / triangleAmount2)),
      b + (r * sin(j * twicePi2 / triangleAmount2))
   );
}
glEnd();



GLfloat c = .15f;
GLfloat d = .85f;
glBegin(GL_TRIANGLE_FAN);
glColor3ub(208, 222, 236);
glVertex2f(c, d); // center of circle
for (j = 0; j <= triangleAmount2; j++)
{
```

```
      glVertex2f(
         c + (r * cos(j * twicePi2 / triangleAmount2)),
         d + (r * sin(j * twicePi2 / triangleAmount2))
      );
   }
   glEnd();



   GLfloat e = .22f;
   GLfloat f = .82f;
   glBegin(GL_TRIANGLE_FAN);
   glColor3ub(208, 222, 236);
   glVertex2f(e, f); // center of circle
   for (j = 0; j <= triangleAmount2; j++)
   {
      glVertex2f(
         e + (r * cos(j * twicePi2 / triangleAmount2)),
         f + (r * sin(j * twicePi2 / triangleAmount2))
      );
   }
   glEnd();



   GLfloat g = .30f;
   GLfloat h = .84f;
   glBegin(GL_TRIANGLE_FAN);
   glColor3ub(208, 222, 236);
   glVertex2f(g, h); // center of circle
   for (j = 0; j <= triangleAmount2; j++)
   {
      glVertex2f(
         g + (r * cos(j * twicePi2 / triangleAmount2)),
         h + (r * sin(j * twicePi2 / triangleAmount2))
      );
   }
   glEnd();

   glPopMatrix();



   glPushMatrix();
   glLoadIdentity();
   glTranslatef(x1Pos, 0.0, 0.0f);
//Bird 1.....................................................................
```

```
   glBegin(GL_QUADS);
   glColor3ub(192,222,72);
   glVertex2f(0.78,0.7875);
   glVertex2f(0.8,0.7875);
   glVertex2f(0.79,0.8);
   glVertex2f(0.77,0.8);
   glEnd();

   glBegin(GL_POLYGON);
   glColor3ub(192,222,72);
   glVertex2f(0.8,0.7875);
   glVertex2f(0.82,0.80);
   glVertex2f(0.83,0.8125);
   glVertex2f(0.82,0.8125);
   glVertex2f(0.81,0.81875);
   glVertex2f(0.8,0.8125);
   glVertex2f(0.79,0.825);
   glVertex2f(0.79,0.8125);
   glVertex2f(0.78,0.825);
   glVertex2f(0.79,0.8);
   glEnd();

//Bird 2.....................................................................
   glBegin(GL_QUADS);
   glColor3ub(51,0,25);
   glVertex2f(0.75,0.85);
glVertex2f(0.77,0.85);
   glVertex2f(0.76,0.8625);
   glVertex2f(0.74,0.8625);
   glEnd();
   glBegin(GL_POLYGON);
   glColor3ub(51,0,25);
   glVertex2f(0.77,0.85);
   glVertex2f(0.79,0.8625);
   glVertex2f(0.8,0.875);
   glVertex2f(0.79,0.875);
   glVertex2f(0.78,0.88125);
   glVertex2f(0.77,0.875);
   glVertex2f(0.76,0.8875);
   glVertex2f(0.76,0.875);
   glVertex2f(0.75,0.8875);
   glVertex2f(0.76,0.8625);
   glEnd();
   glLoadIdentity();
   glPopMatrix();

   glutSwapBuffers();
}
```

```
void handleSpecialKeypress(int key, int x, int y)
{
   switch(key)
   {
   case GLUT_KEY_LEFT:
      translateX -= 0.01f;
      break;
   case GLUT_KEY_RIGHT:
      translateX += 0.01f;
      break;
   case GLUT_KEY_UP:
      translateY += 0.01f;
      break;
   case GLUT_KEY_DOWN:
      translateY -= 0.01f;
      break;
   }
   glutPostRedisplay();
}
```

```
void keyboard(unsigned char key, int x, int y)
{
   switch (key)
   {
   //boat3 move.....................

   case 'a':
      xPos -= 0.01f;
      break;
   case 'd':
      xPos += 0.01f;
      break;

  //boat2 move.....................

   case 'o':
```

```
        xP -= 0.01f;
        break;
    case 'p':
        xP += 0.01f;
        break;


    //star move.....................

    case 's':
        yPos -= 0.01f;
        break;
    case 'w':
        yPos += 0.01f;
        break;

    //bg colour.................

    case 'z':
        p = 51,q=204,f=255;
        break;

    case 'x':
        p = 255,q=204,f=0;
        break;

    case 'c':
        p = 0,q=0,f=0;
        break;


    //sun colour.....................................

    case ',':
        u = 255,v=204,w=0;
        break;

    case '.':
        u = 1,v=1,w=1;
        break;

    default:
        break;
    }
    glutPostRedisplay();
}
```

```
void update(int value)
{
   xPos += 0.01f;

   if (xPos > 1)
   {
      xPos = 0 - xPos;
   }


   glutPostRedisplay();
   glutTimerFunc(40, update, 0);
}
void update1(int value)
{
   xP -= 0.01f;

   if (xP < -1)
   {
      xP = 0-xP ;
   }


   glutPostRedisplay();
   glutTimerFunc(40, update1, 0);
}



void update2(int value)
{
   x1Pos += 0.01f;

   if (x1Pos > 1)
   {
      x1Pos = 0 - x1Pos;
   }


   glutPostRedisplay();
   glutTimerFunc(60, update2, 0);
}
```

```
void handleMouse(int button, int state, int x, int y)
{
   if (button == GLUT_LEFT_BUTTON && state == GLUT_DOWN)
   {
      angle += 10.0f;
      if (angle > 360.0f)
         angle -= 360.0f;
      glutPostRedisplay();
   }
}




void update3(int value)
{
   angle -= 5.0f;
   if (angle > 360.0f)
      angle -= 360.0f;
   glutPostRedisplay();
   glutTimerFunc(25, update3, 0);
}



int main(int argc, char **argv)
{
   glutInit(&argc, argv);
   glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
   glutInitWindowSize(400, 400);
   glutCreateWindow("Moving Square");
   glClearColor(1.0f, 1.0f, 1.0f, 1.0f);  // White background
   glMatrixMode(GL_PROJECTION);
   glLoadIdentity();
   gluOrtho2D(0.0f, 1.0f, 0.0f, 1.0f);
   glMatrixMode(GL_MODELVIEW);
   glutDisplayFunc(display);
   glutKeyboardFunc(keyboard);
   glutSpecialFunc(handleSpecialKeypress);
   glutMouseFunc(handleMouse);
   glutTimerFunc(40, update, 0);
   glutTimerFunc(40, update1, 0);
   glutTimerFunc(60, update2, 0);
   glutTimerFunc(25, update3, 0);
   glutMainLoop();
   return 0;
}
```

Screen Shot: