Department of Computer Science and Engineering

Final Project

Project Title:

Forest with Modern Objects

Name: Abdultawwab Safarji (3710933)

Submited to:

Mr.Basim Iskandarani

Computer Graphics (CS-376)

Apr 15, 2020

1 Problem Statement

There are many ideas to put into imagination, but I would like to apply all of what I have learned so far. So, I would like to make a wide area of grean trees and using the helicopter that we have created and make it more controllable. I wanted to bulid houses and area for parking the helicopter as well. I will used all the graphics featuers such as texture, animation, lighting, and 3D objects.

2 Objective of the Project

The goal of this project is to create big field of green area, houses and place for parking the helicopter as well. I will used all the graphics featuers such as texture, animation, lighting, and 3D objects.

.

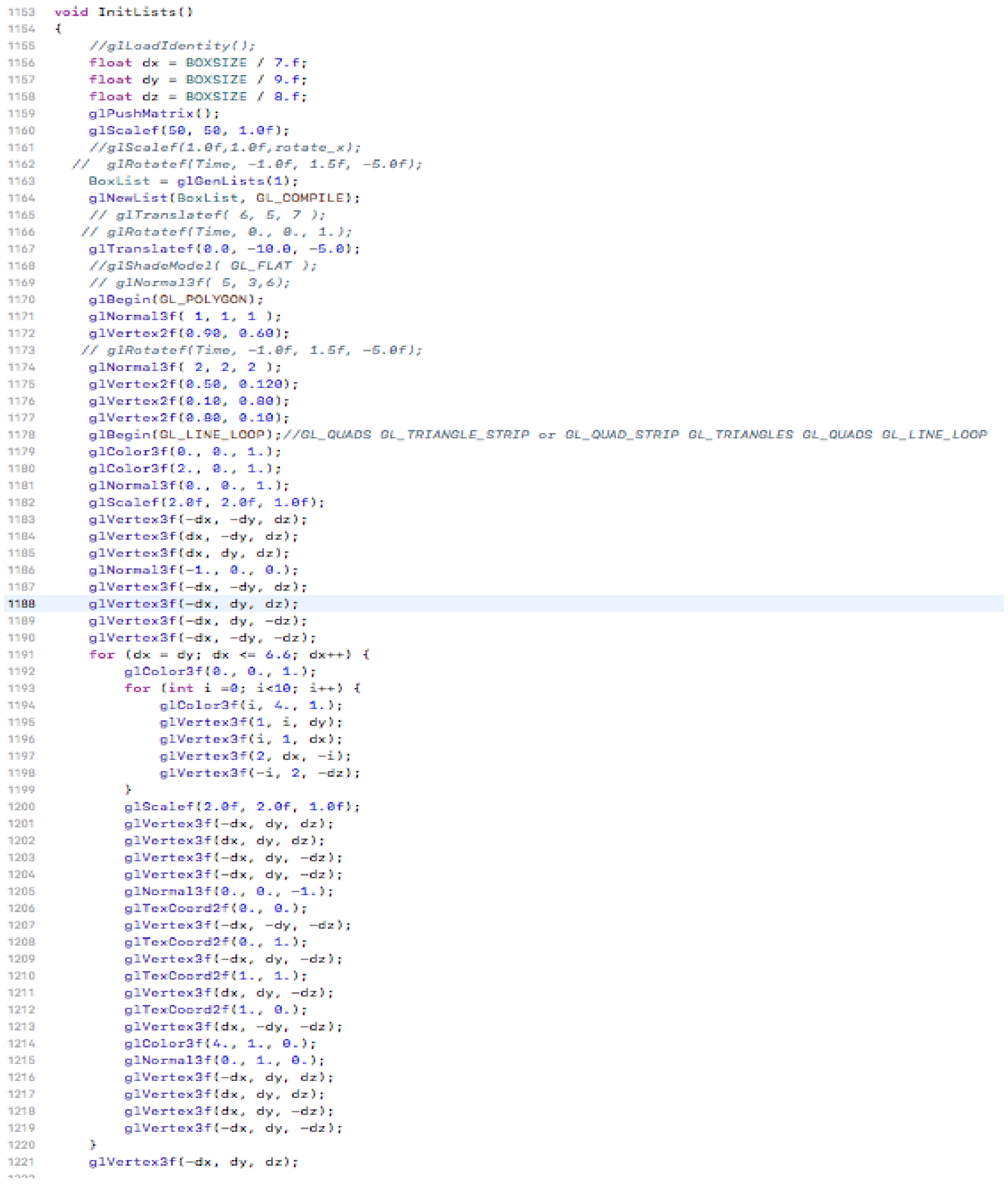
3 Functionalities

In order to achieve the goal of the project is to use the following functions,

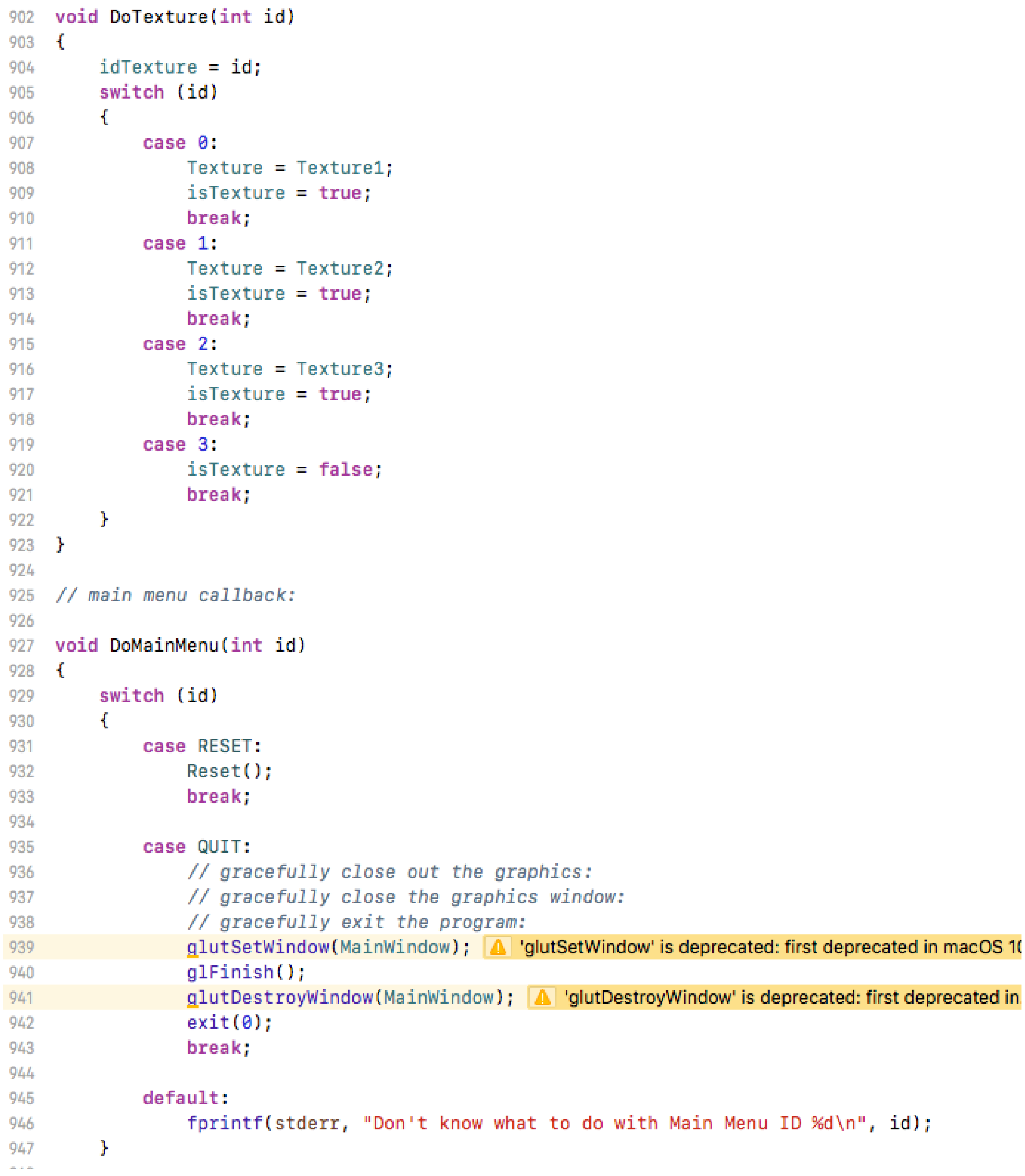
- Lighting
- Textuer
- Coloring
- Drawing vertexes 3D
- Animation
- 3D objects

Drawing an example

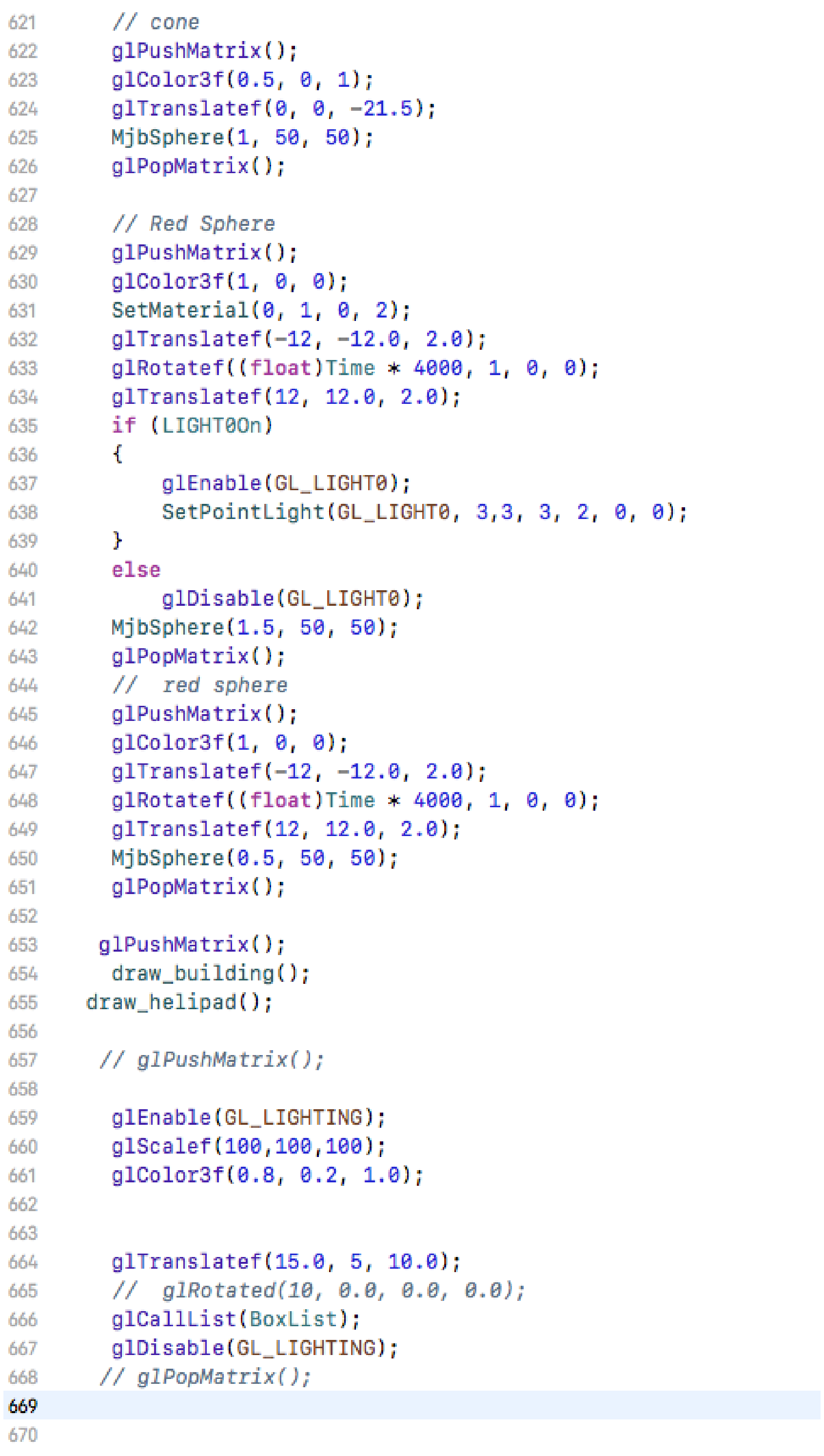
4 Code



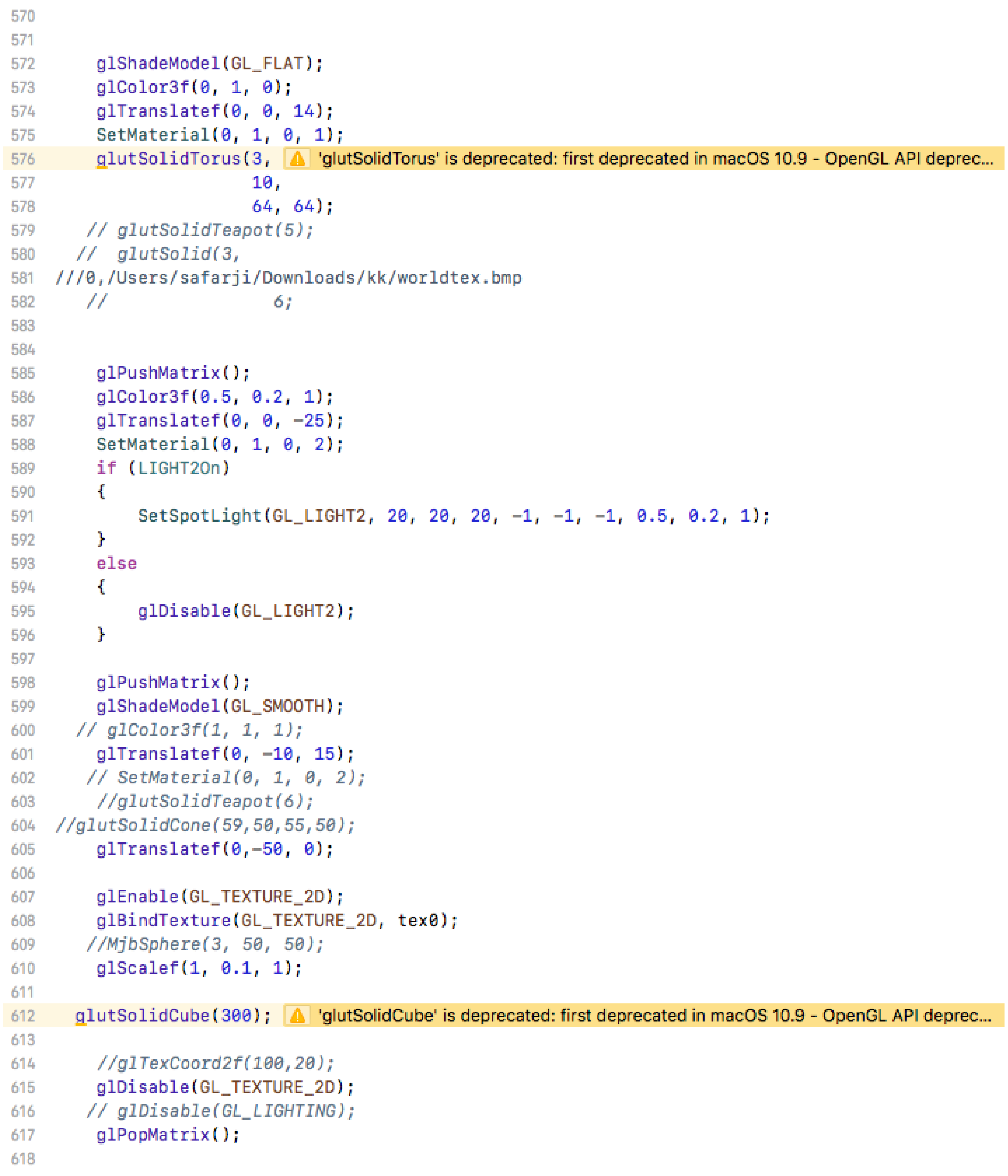
```c
void InitLists()
{
    //glLoadIdentity();
    float dx = BOXSIZE / 7.f;
    float dy = BOXSIZE / 9.f;
    float dz = BOXSIZE / 8.f;
    glPushMatrix();
    glScalef(50, 50, 1.0f);
    //glScalef(1.0f,1.0f,rotate_x);
  //  glRotatef(Time, -1.0f, 1.5f, -5.0f);
    BoxList = glGenLists(1);
    glNewList(BoxList, GL_COMPILE);
    // glTranslatef( 6, 5, 7 );
   // glRotatef(Time, 0., 0., 1.);
    glTranslatef(0.0, -10.0, -5.0);
    //glShadeModel( GL_FLAT );
    // glNormal3f( 5, 3,6);
    glBegin(GL_POLYGON);
    glNormal3f( 1, 1, 1 );
    glVertex2f(0.90, 0.60);
   // glRotatef(Time, -1.0f, 1.5f, -5.0f);
    glNormal3f( 2, 2, 2 );
    glVertex2f(0.50, 0.120);
    glVertex2f(0.10, 0.80);
    glVertex2f(0.80, 0.10);
    glBegin(GL_LINE_LOOP);//GL_QUADS GL_TRIANGLE_STRIP or GL_QUAD_STRIP GL_TRIANGLES GL_QUADS GL_LINE_LOOP
    glColor3f(0., 0., 1.);
    glColor3f(2., 0., 1.);
    glNormal3f(0., 0., 1.);
    glScalef(2.0f, 2.0f, 1.0f);
    glVertex3f(-dx, -dy, dz);
    glVertex3f(dx, -dy, dz);
    glVertex3f(dx, dy, dz);
    glNormal3f(-1., 0., 0.);
    glVertex3f(-dx, -dy, dz);
    glVertex3f(-dx, dy, dz);
    glVertex3f(-dx, dy, -dz);
    glVertex3f(-dx, -dy, -dz);
    for (dx = dy; dx <= 6.6; dx++) {
        glColor3f(0., 0., 1.);
        for (int i =0; i<10; i++) {
            glColor3f(i, 4., 1.);
            glVertex3f(1, i, dy);
            glVertex3f(i, 1, dx);
            glVertex3f(2, dx, -i);
            glVertex3f(-i, 2, -dz);
        }
        glScalef(2.0f, 2.0f, 1.0f);
        glVertex3f(-dx, dy, dz);
        glVertex3f(dx, dy, dz);
        glVertex3f(-dx, dy, -dz);
        glVertex3f(-dx, dy, -dz);
        glNormal3f(0., 0., -1.);
        glTexCoord2f(0., 0.);
        glVertex3f(-dx, -dy, -dz);
        glTexCoord2f(0., 1.);
        glVertex3f(-dx, dy, -dz);
        glTexCoord2f(1., 1.);
        glVertex3f(dx, dy, -dz);
        glTexCoord2f(1., 0.);
        glVertex3f(dx, -dy, -dz);
        glColor3f(4., 1., 0.);
        glNormal3f(0., 1., 0.);
        glVertex3f(-dx, dy, dz);
        glVertex3f(dx, dy, dz);
        glVertex3f(dx, dy, -dz);
        glVertex3f(-dx, dy, -dz);
    }
    glVertex3f(-dx, dy, dz);
```



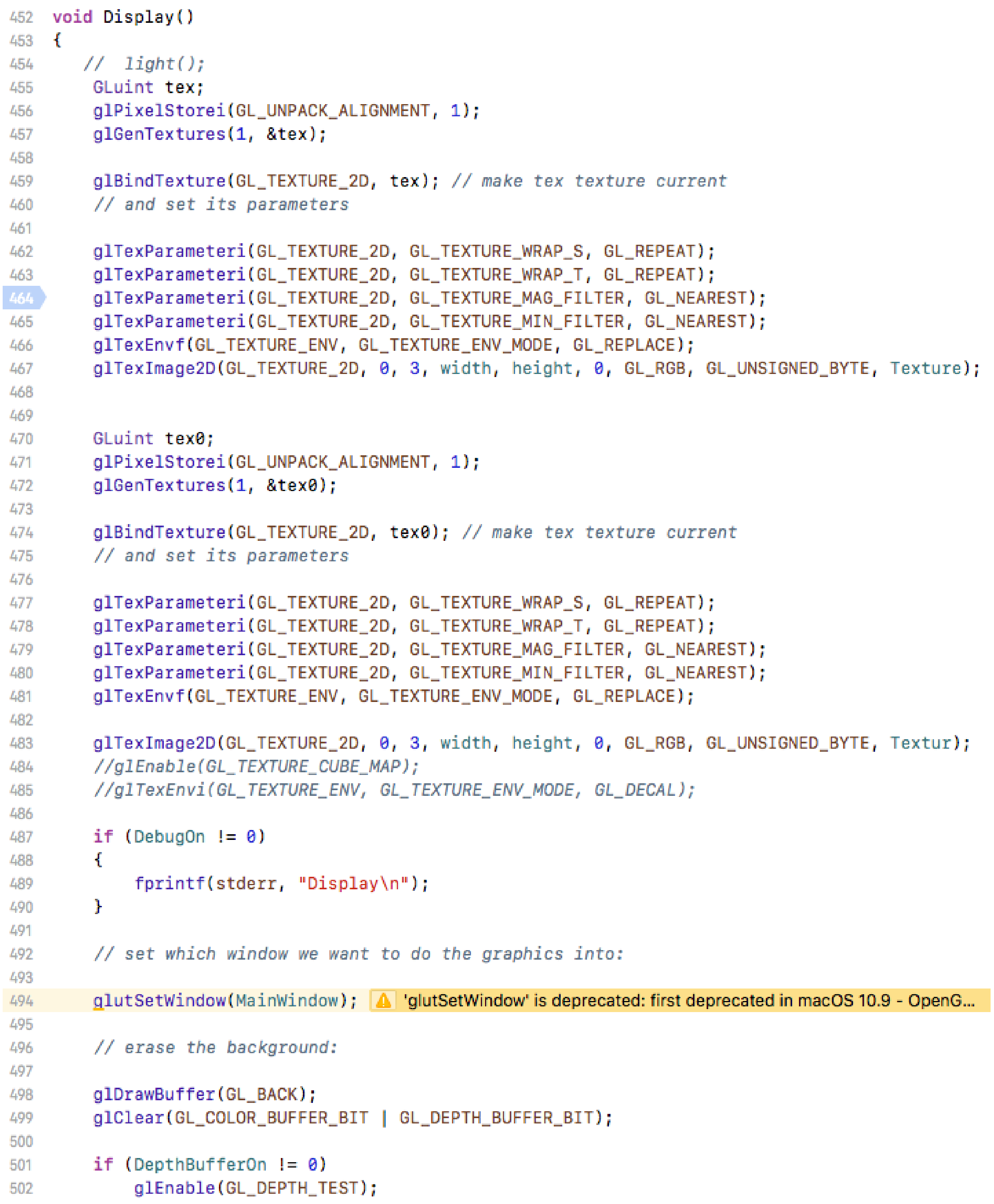
```
void DoTexture(int id)
{
    idTexture = id;
    switch (id)
    {
        case 0:
            Texture = Texture1;
            isTexture = true;
            break;
        case 1:
            Texture = Texture2;
            isTexture = true;
            break;
        case 2:
            Texture = Texture3;
            isTexture = true;
            break;
        case 3:
            isTexture = false;
            break;
    }
}

// main menu callback:

void DoMainMenu(int id)
{
    switch (id)
    {
        case RESET:
            Reset();
            break;

        case QUIT:
            // gracefully close out the graphics:
            // gracefully close the graphics window:
            // gracefully exit the program:
            glutSetWindow(MainWindow);  ⚠ 'glutSetWindow' is deprecated: first deprecated in macOS 1
            glFinish();
            glutDestroyWindow(MainWindow);  ⚠ 'glutDestroyWindow' is deprecated: first deprecated in
            exit(0);
            break;

        default:
            fprintf(stderr, "Don't know what to do with Main Menu ID %d\n", id);
    }
```



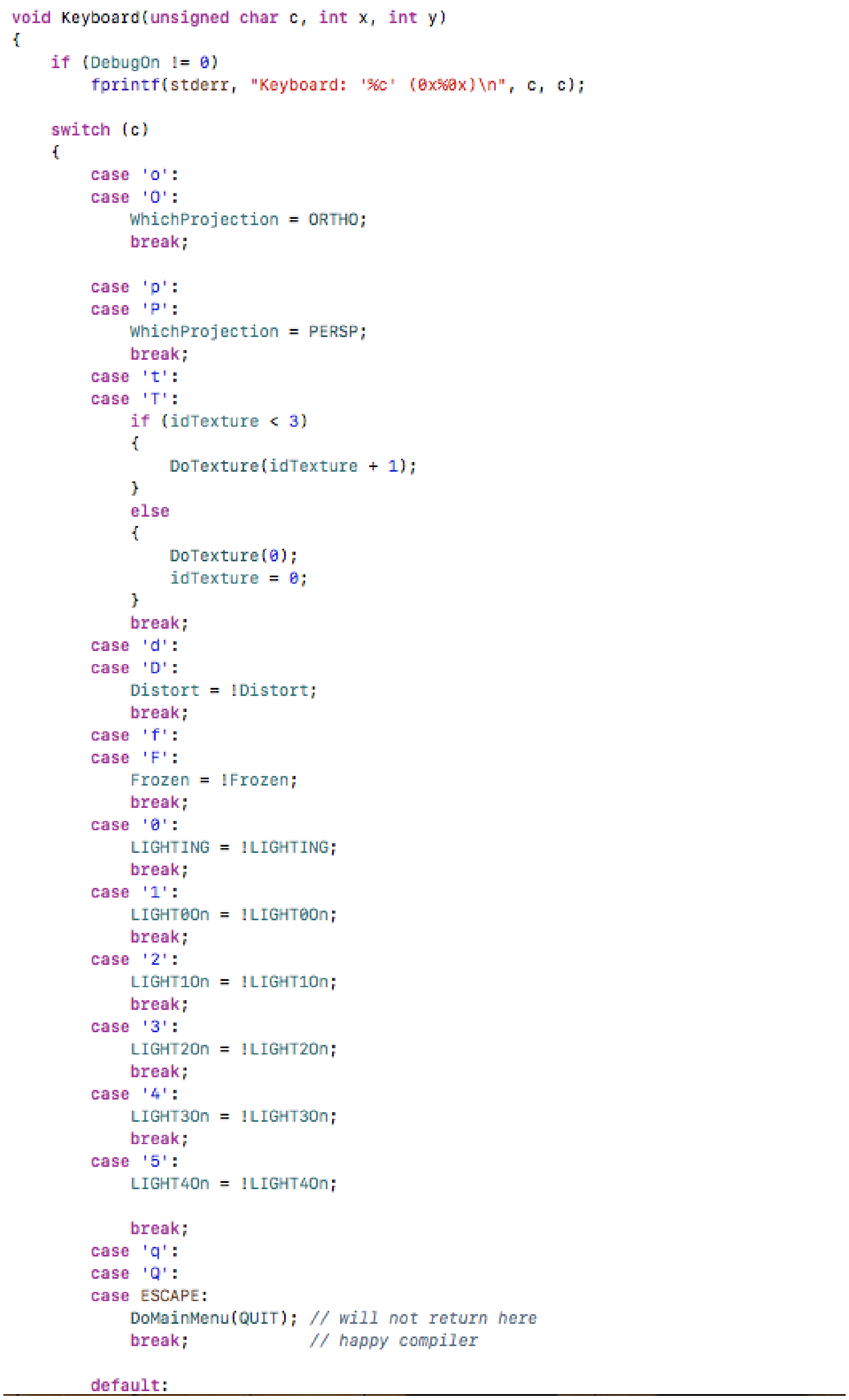
```
    // cone
    glPushMatrix();
    glColor3f(0.5, 0, 1);
    glTranslatef(0, 0, -21.5);
    MjbSphere(1, 50, 50);
    glPopMatrix();

    // Red Sphere
    glPushMatrix();
    glColor3f(1, 0, 0);
    SetMaterial(0, 1, 0, 2);
    glTranslatef(-12, -12.0, 2.0);
    glRotatef((float)Time * 4000, 1, 0, 0);
    glTranslatef(12, 12.0, 2.0);
    if (LIGHT0On)
    {
        glEnable(GL_LIGHT0);
        SetPointLight(GL_LIGHT0, 3,3, 3, 2, 0, 0);
    }
    else
        glDisable(GL_LIGHT0);
    MjbSphere(1.5, 50, 50);
    glPopMatrix();
    //  red sphere
    glPushMatrix();
    glColor3f(1, 0, 0);
    glTranslatef(-12, -12.0, 2.0);
    glRotatef((float)Time * 4000, 1, 0, 0);
    glTranslatef(12, 12.0, 2.0);
    MjbSphere(0.5, 50, 50);
    glPopMatrix();

   glPushMatrix();
    draw_building();
  draw_helipad();

   // glPushMatrix();

    glEnable(GL_LIGHTING);
    glScalef(100,100,100);
    glColor3f(0.8, 0.2, 1.0);


    glTranslatef(15.0, 5, 10.0);
    //  glRotated(10, 0.0, 0.0, 0.0);
    glCallList(BoxList);
    glDisable(GL_LIGHTING);
   // glPopMatrix();



   glPopMatrix();
  glPushMatrix();
    glTranslatef(-40, 0, -50);
draw_house();
 glPopMatrix();
 // Blue Sphere
 glPushMatrix();
 glColor3f(0, 0, 1);
 SetMaterial(0, 1, 0, 2);
 glTranslatef(0, 0, -14.0);
 glTranslatef(10, 10.0, -10.0);
 glRotatef((float)Time * 4000, 1, 0, 0);
 glTranslatef(-10, -10.0, -10.0);
 if (LIGHT1On)
 {
     glEnable(GL_LIGHT1);
     SetPointLight(GL_LIGHT1, 2, 2, 2, 0, 0, 1);
 }
 else
     glDisable(GL_LIGHT1);
 MjbSphere(1.5, 50, 50);
 glPopMatrix();
 // Small  the blu sphere
 glPushMatrix();
 glColor3f(0, 0, 1);
 glTranslatef(0, 0, -14.0);
 glTranslatef(10, 10.0, -10.8);
 glRotatef((float)Time * 4000, 1, 0, 0);
 glTranslatef(-10, -10.0, -10.8);
 MjbSphere(0.5, 50, 50);
 glPopMatrix();

 // S texture
 glPushMatrix();
 glShadeModel(GL_SMOOTH);

 glColor3f(1, 1, 1);
 glTranslatef(0, -10, 15);
 SetMaterial(0, 1, 0, 2);
 if (isTexture)
 {
     glEnable(GL_TEXTURE_2D);
     glBindTexture(GL_TEXTURE_2D, tex);
    MjbSphere(1000, 300, 300);

    // glTexCoord2f(100,20);
     glDisable(GL_TEXTURE_2D);
 }
```



```



    glShadeModel(GL_FLAT);
    glColor3f(0, 1, 0);
    glTranslatef(0, 0, 14);
    SetMaterial(0, 1, 0, 1);
    glutSolidTorus(3,  ⚠ 'glutSolidTorus' is deprecated: first deprecated in macOS 10.9 - OpenGL API deprec...
                   10,
                   64, 64);
   // glutSolidTeapot(5);
  //  glutSolid(3,
///0,/Users/safarji/Downloads/kk/worldtex.bmp
   //                6;


    glPushMatrix();
    glColor3f(0.5, 0.2, 1);
    glTranslatef(0, 0, -25);
    SetMaterial(0, 1, 0, 2);
    if (LIGHT2On)
    {
        SetSpotLight(GL_LIGHT2, 20, 20, 20, -1, -1, -1, 0.5, 0.2, 1);
    }
    else
    {
        glDisable(GL_LIGHT2);
    }

    glPushMatrix();
    glShadeModel(GL_SMOOTH);
  // glColor3f(1, 1, 1);
    glTranslatef(0, -10, 15);
   // SetMaterial(0, 1, 0, 2);
    //glutSolidTeapot(6);
//glutSolidCone(59,50,55,50);
    glTranslatef(0,-50, 0);

    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, tex0);
   //MjbSphere(3, 50, 50);
    glScalef(1, 0.1, 1);

  glutSolidCube(300);  ⚠ 'glutSolidCube' is deprecated: first deprecated in macOS 10.9 - OpenGL API deprec...

    //glTexCoord2f(100,20);
    glDisable(GL_TEXTURE_2D);
   // glDisable(GL_LIGHTING);
    glPopMatrix();

```



```
void Display()
{
   //  light();
    GLuint tex;
    glPixelStorei(GL_UNPACK_ALIGNMENT, 1);
    glGenTextures(1, &tex);

    glBindTexture(GL_TEXTURE_2D, tex); // make tex texture current
    // and set its parameters

    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_NEAREST);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
    glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_REPLACE);
    glTexImage2D(GL_TEXTURE_2D, 0, 3, width, height, 0, GL_RGB, GL_UNSIGNED_BYTE, Texture);


    GLuint tex0;
    glPixelStorei(GL_UNPACK_ALIGNMENT, 1);
    glGenTextures(1, &tex0);

    glBindTexture(GL_TEXTURE_2D, tex0); // make tex texture current
    // and set its parameters

    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_NEAREST);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
    glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_REPLACE);

    glTexImage2D(GL_TEXTURE_2D, 0, 3, width, height, 0, GL_RGB, GL_UNSIGNED_BYTE, Textur);
    //glEnable(GL_TEXTURE_CUBE_MAP);
    //glTexEnvi(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_DECAL);

    if (DebugOn != 0)
    {
        fprintf(stderr, "Display\n");
    }

    // set which window we want to do the graphics into:

    glutSetWindow(MainWindow);   ⚠ 'glutSetWindow' is deprecated: first deprecated in macOS 10.9 - OpenG...

    // erase the background:

    glDrawBuffer(GL_BACK);
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    if (DepthBufferOn != 0)
        glEnable(GL_DEPTH_TEST);
```

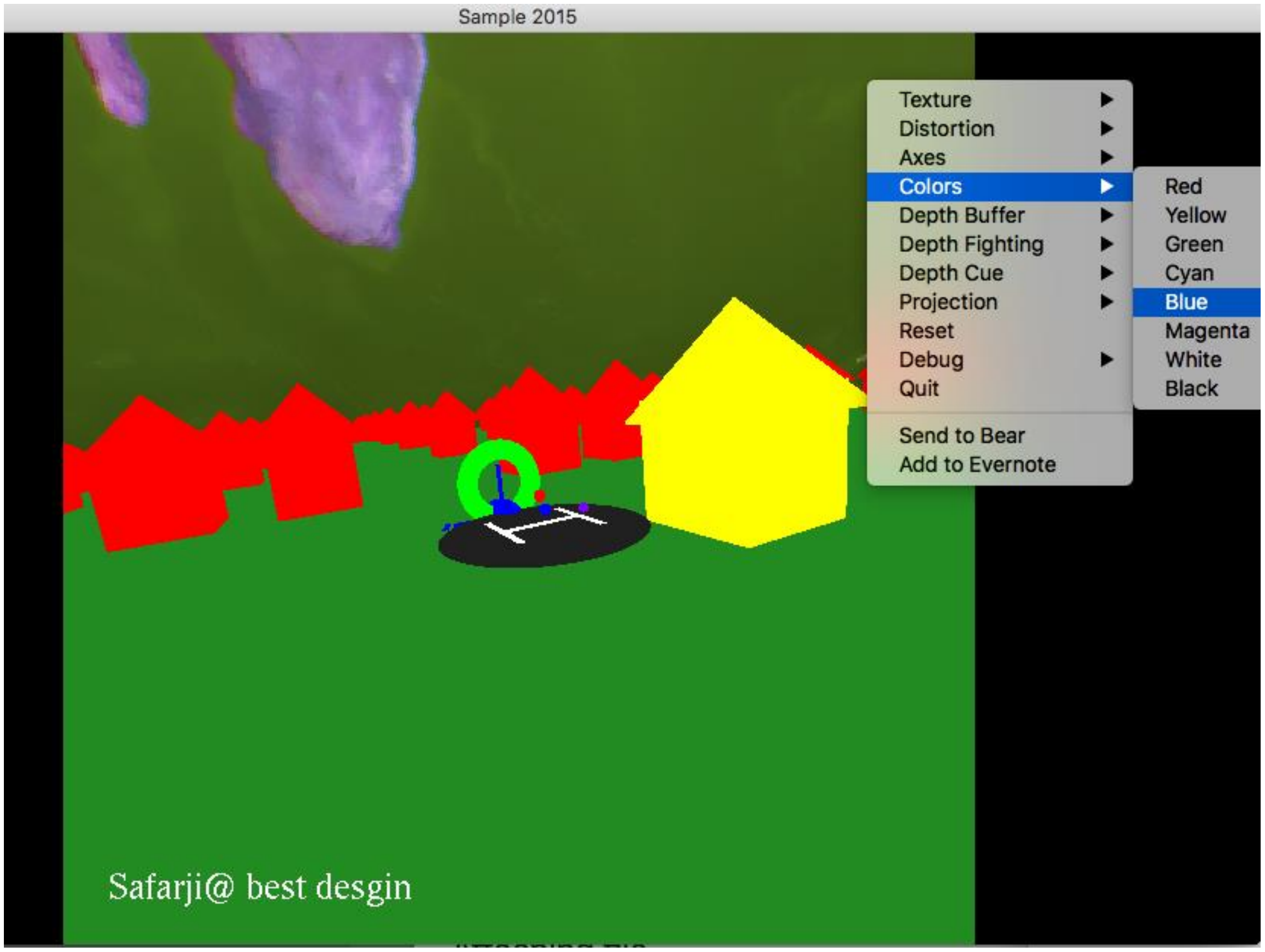


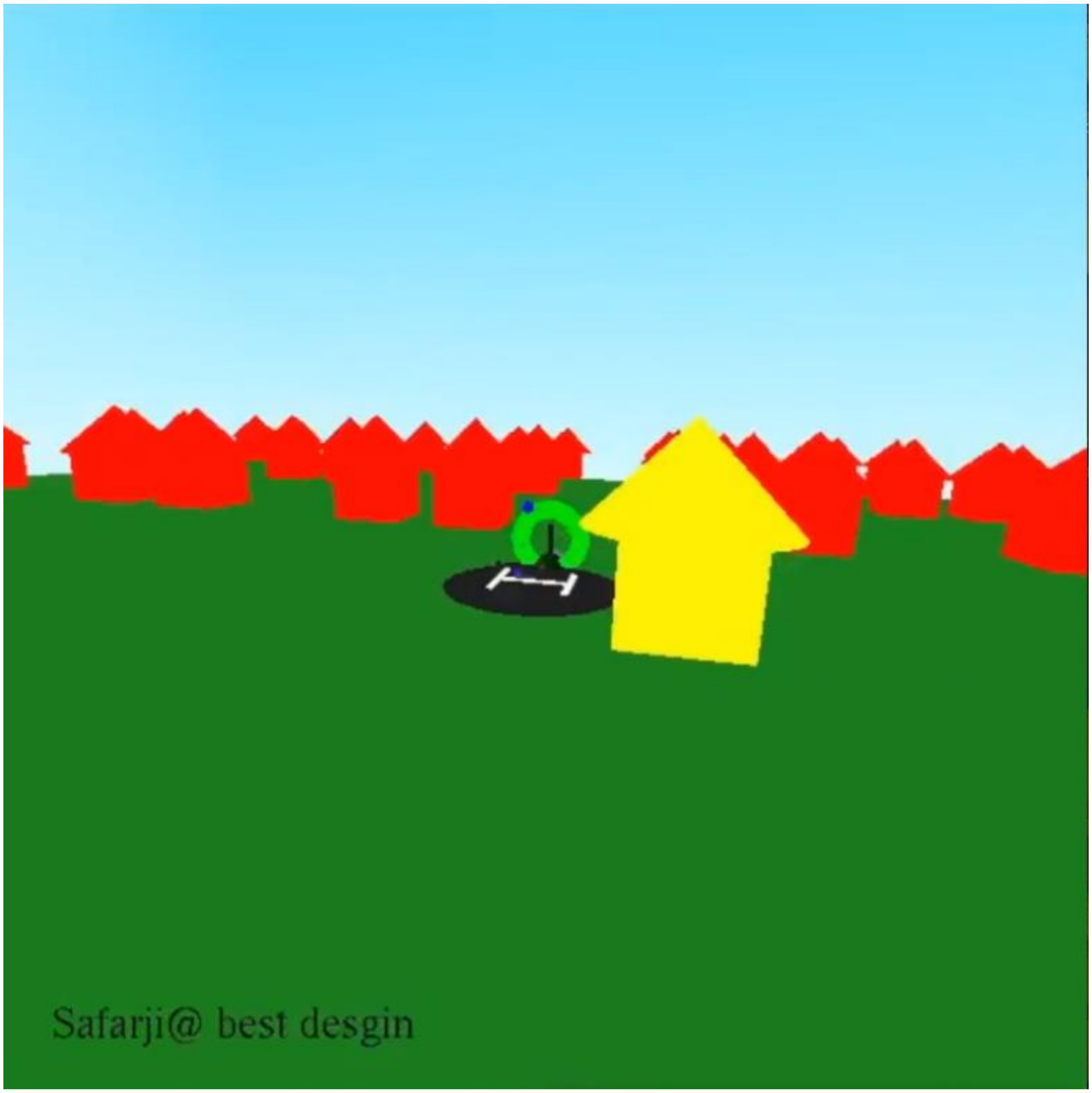
```
void Keyboard(unsigned char c, int x, int y)
{
    if (DebugOn != 0)
        fprintf(stderr, "Keyboard: '%c' (0x%0x)\n", c, c);

    switch (c)
    {
        case 'o':
        case 'O':
            WhichProjection = ORTHO;
            break;

        case 'p':
        case 'P':
            WhichProjection = PERSP;
            break;
        case 't':
        case 'T':
            if (idTexture < 3)
            {
                DoTexture(idTexture + 1);
            }
            else
            {
                DoTexture(0);
                idTexture = 0;
            }
            break;
        case 'd':
        case 'D':
            Distort = !Distort;
            break;
        case 'f':
        case 'F':
            Frozen = !Frozen;
            break;
        case '0':
            LIGHTING = !LIGHTING;
            break;
        case '1':
            LIGHT0On = !LIGHT0On;
            break;
        case '2':
            LIGHT1On = !LIGHT1On;
            break;
        case '3':
            LIGHT2On = !LIGHT2On;
            break;
        case '4':
            LIGHT3On = !LIGHT3On;
            break;
        case '5':
            LIGHT4On = !LIGHT4On;

            break;
        case 'q':
        case 'Q':
        case ESCAPE:
            DoMainMenu(QUIT); // will not return here
            break;            // happy compiler

        default:
```

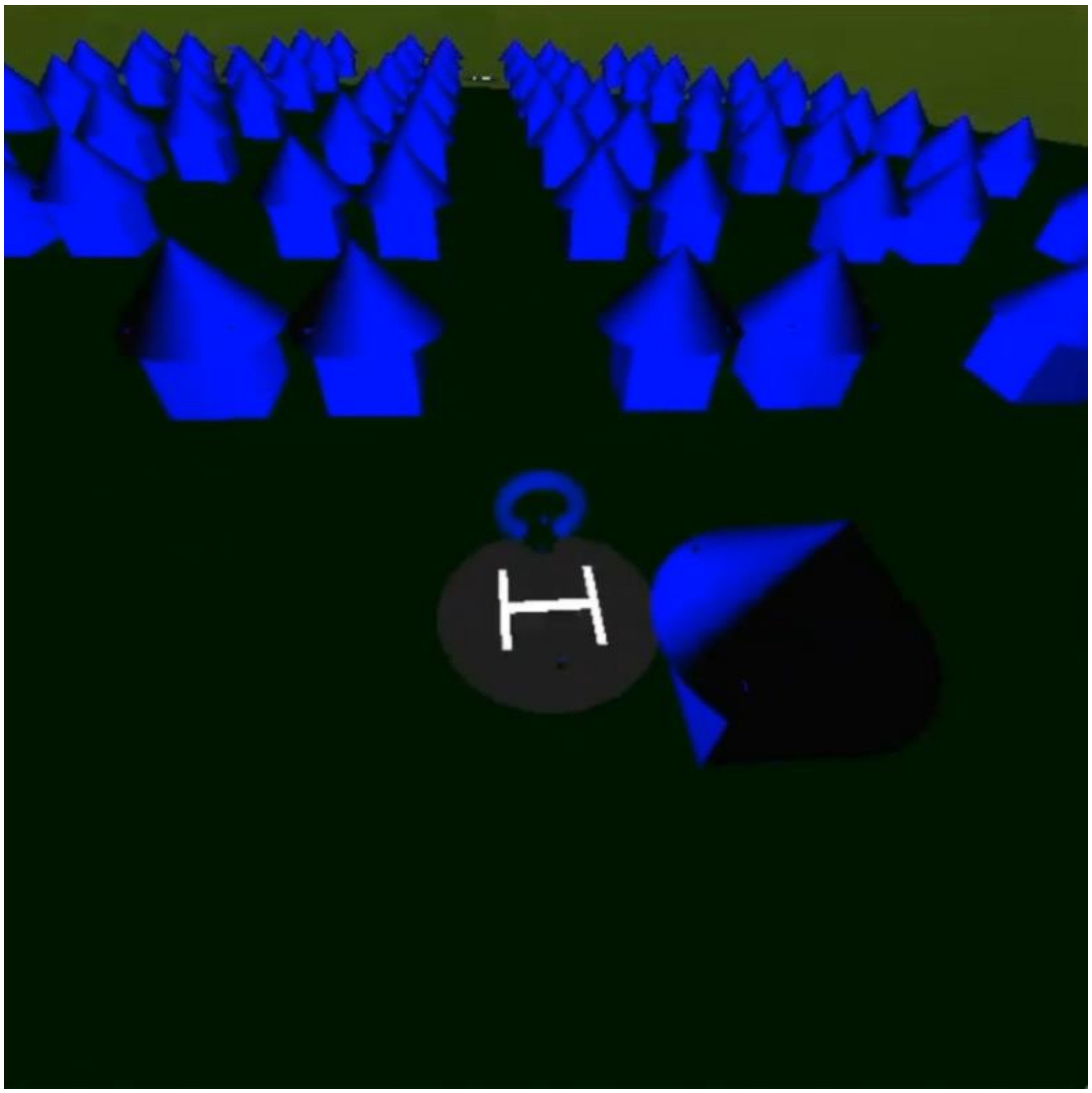
5 Demo

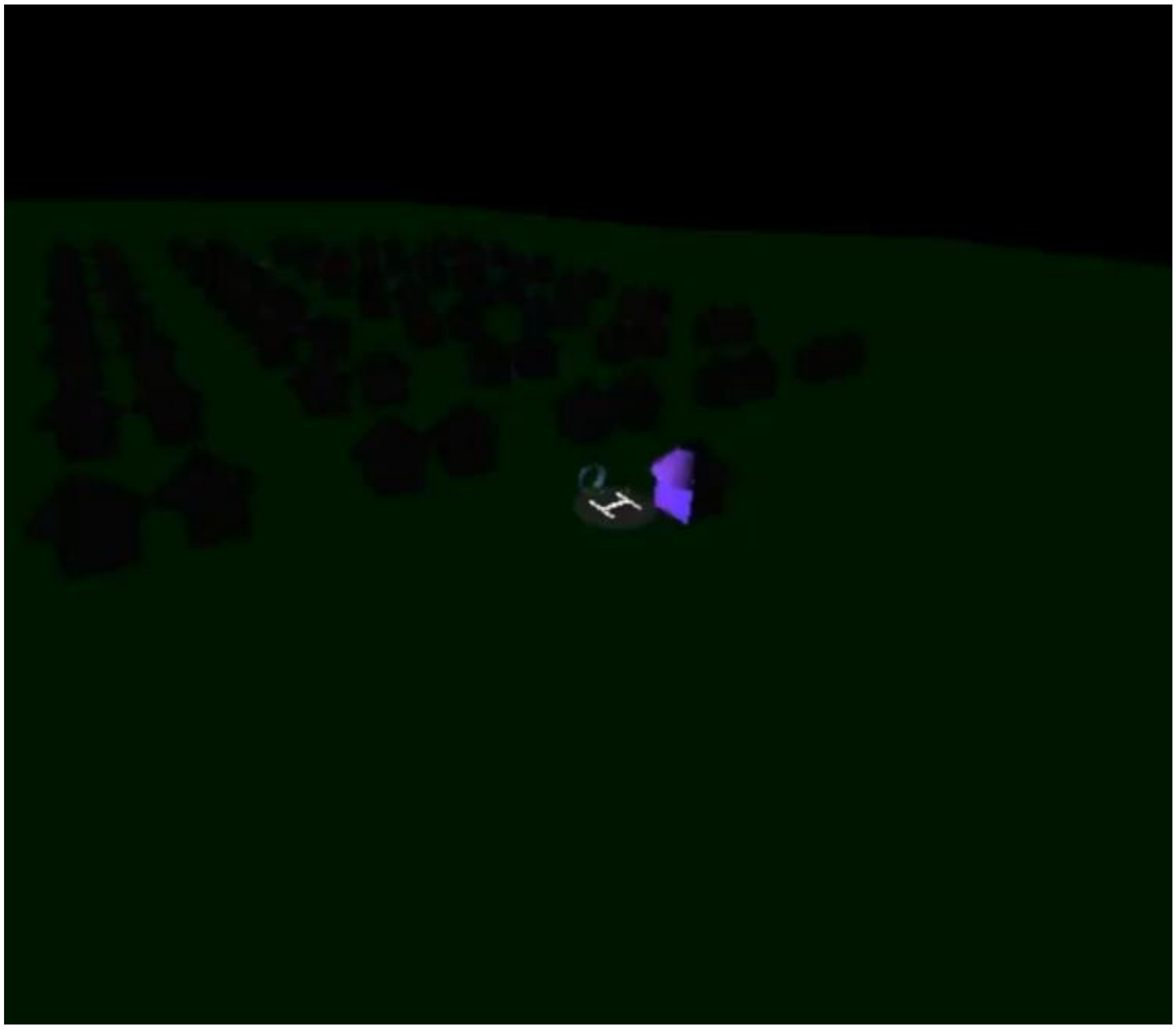
Tips:
D= distortion
P= prospect
zero= lightings off
1 to 4=lights and shadow
T= multi textuers
5= sky mapping
F= freeze animation

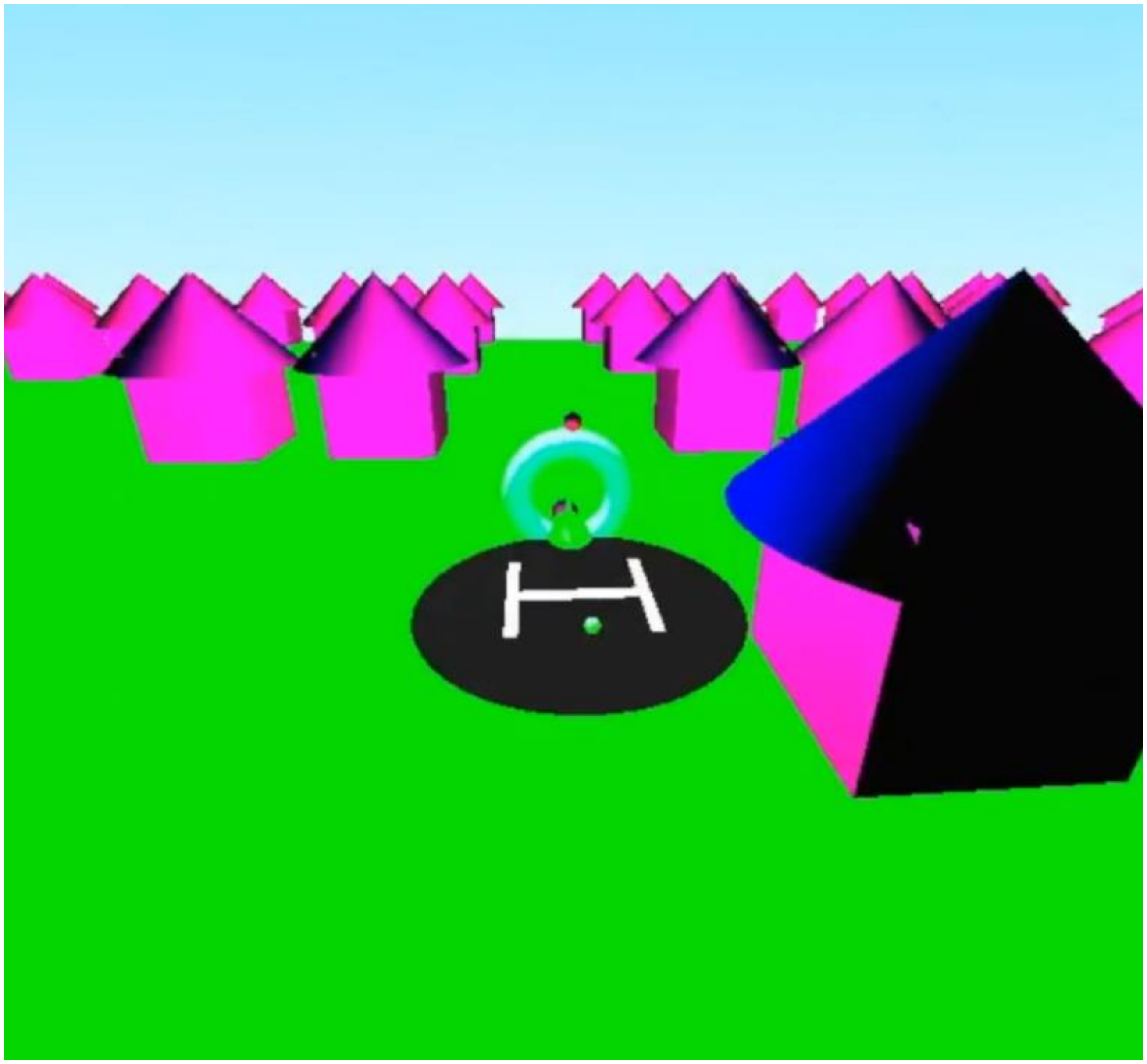
change to multiple colors by clicking the menu and choose the color.




Safarji@ best desgin


H


H


H

Safarji@ best desgin