Experiment no : 06

Name : Atharva B. Iparkar   Date : 17/10/2024

Roll no : S211045

Class : S.E.

Div : A

Batch : A-2

Problem Statement :

Write OpenGL program to draw Sun Rise and Sunset.

Code :

```
#include<iostream>

#include<stdlib.h>

#include<GL/glut.h>

using namespace std;

float ballX = -0.8f;

float ballY = -0.3f;

float ballZ = -1.2f;

float colR=3.0;

float colG=1.5;

float colB=1.0;

float bgColR=0.0;

float bgColG=0.0;
float bgColB=0.0;

static int flag=1;


void drawBall(void) {

glColor3f(colR,colG,col

B);

glTranslatef(ballX,ballY,

ballZ);

glutSolidSphere (0.05,

30, 30);

}


void drawAv(void) {

glBegin(GL_POLYGON

);

glColor3f(1.0,1.0,1.0);

glVertex3f(-0.9,-0.7,-

1.0);

glVertex3f(-0.5,-0.1,-

1.0);
glVertex3f(-0.2,-1.0,-
1.0);
glVertex3f(0.5,0.0,-1.0);
glVertex3f(0.6,-0.2,-1.0);
glVertex3f(0.9,-0.7,-1.0);
glEnd();
}

void initRendering() {
glEnable(GL_DEPTH_T
EST);
glEnable(GL_COLOR_
MATERIAL);
glEnable(GL_LIGHTIN
G);
glEnable(GL_LIGHT0);
glEnable(GL_LIGHT1);
glEnable(GL_NORMAL
IZE);
}
```

```
void handleResize(int w,
int h) {
glViewport(0, 0, w, h);
glMatrixMode(GL_PRO
JECTION);
glLoadIdentity();
gluPerspective(45.0,(dou
ble)w / (double)h, 1.0,
200.0);
}

void drawScene() {
glClear(GL_COLOR_B
UFFER_BIT|GL_DEPT
H_BUFFER_BIT);
glClearColor(bgColR,bg
ColG,bgColB,0.0);
glMatrixMode(GL_MO
DELVIEW);
```

```
glLoadIdentity();

GLfloat ambientColor[]

= {0.2f, 0.2f, 0.2f, 1.0f};

glLightModelfv(GL_LI

GHT_MODEL_AMBIE

NT, ambientColor);

GLfloat lightColor0[] =

{0.5f, 0.5f, 0.5f, 1.0f};

GLfloat lightPos0[] =

{4.0f, 0.0f, 8.0f, 1.0f};

glLightfv(GL_LIGHT0,

GL_DIFFUSE,

lightColor0);

glLightfv(GL_LIGHT0,

GL_POSITION,

lightPos0);

GLfloat lightColor1[] =

{0.5f, 0.2f, 0.2f, 1.0f};

GLfloat lightPos1[] = {-

1.0f, 0.5f, 0.5f, 0.0f};
```

```
glLightfv(GL_LIGHT1,

GL_DIFFUSE,

lightColor1);

glLightfv(GL_LIGHT1,

GL_POSITION,

lightPos1);

glPushMatrix();

drawBall();

glPopMatrix();

glPushMatrix();

drawAv();

glPopMatrix();

glPushMatrix();

glPopMatrix();

glutSwapBuffers();

}


void update(int value) {

if(ballX>0.9f) {

ballX = -0.8f;
```

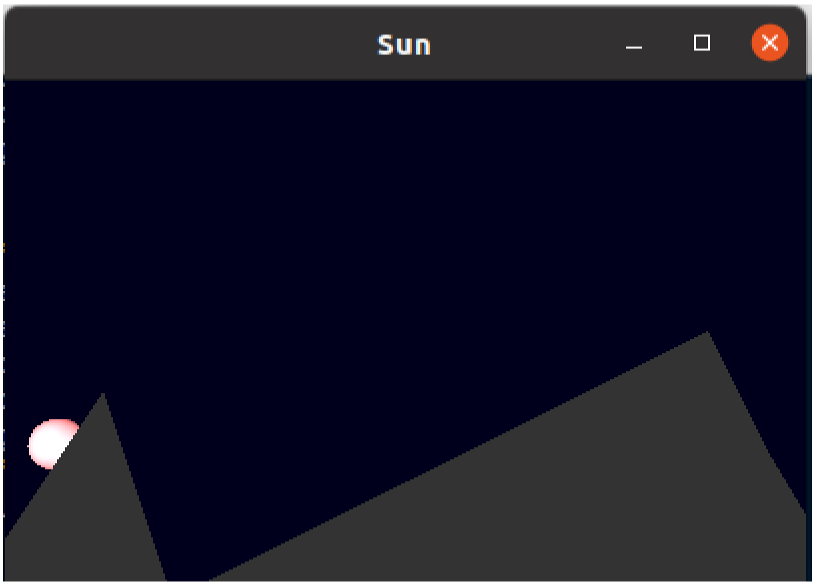
```
ballY = -0.3f;

flag=1;

colR=2.0;

colG=1.50;

colB=1.0;

bgColB=0.0;

}


if(flag) {

ballX += 0.001f;

ballY +=0.0007f;

colR-=0.001;

colB+=0.005;

bgColB+=0.001;

if(ballX>0.01) {

flag=0;

}

}


if (!flag) {
```

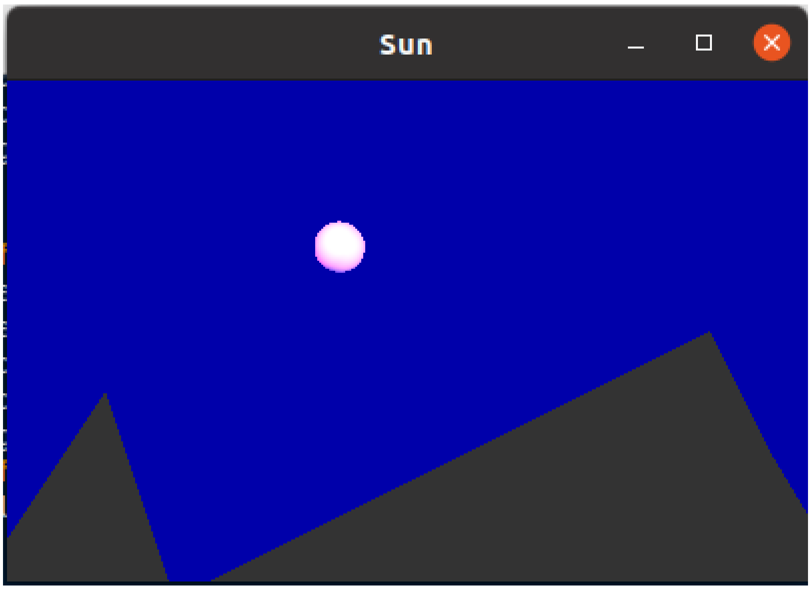
```
ballX += 0.001f;
ballY -=0.0007f;
colR+=0.001;
colB-=0.01;
bgColB-=0.001;
if(ballX<-0.3) {
flag=1;
}
}

glutPostRedisplay();
glutTimerFunc(25,
update, 0);
}

int main(int argc,char**
argv) {
glutInit(&argc,argv);
```

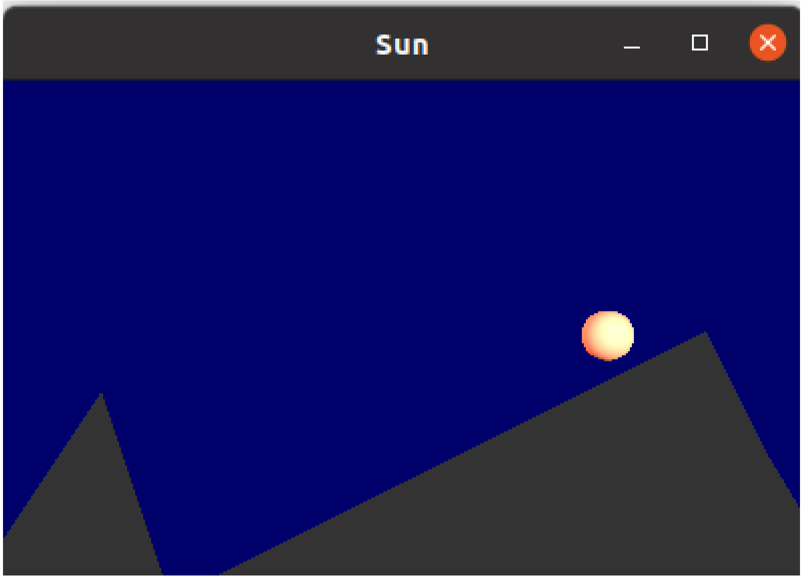
```
glutInitDisplayMode(GL
UT_DOUBLE|GLUT_R
GB|GLUT_DEPTH);
glutInitWindowSize(400
,250);
glutCreateWindow("Sun
");
initRendering();
glutDisplayFunc(drawSc
ene);
glutReshapeFunc(handle
Resize);
glutTimerFunc(250,
update, 0);
glutMainLoop();
return 0;
}
```

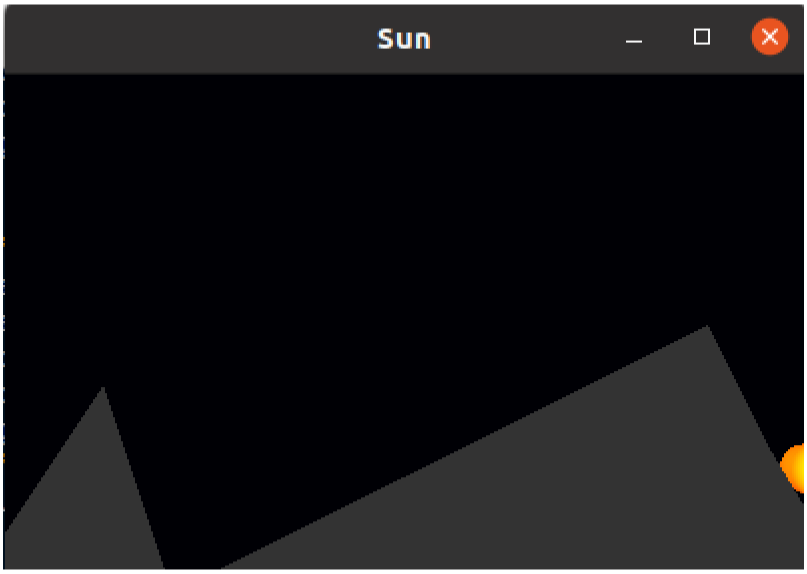
Output :

```
d_comp_pl_ii_11@d-comp-pl-ii-11:~/SE_A2_S211045_Atharva$ g++ SunRise_SunSet.cpp -o s -lGL -lGLU -lglut
d_comp_pl_ii_11@d-comp-pl-ii-11:~/SE_A2_S211045_Atharva$ ./s
```


Sun


Sun


Sun


Sun