

ГУАП

КАФЕДРА № 43

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

должность, уч. степень, звание

подпись, дата

Щёкин С. В.

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

Работа с источником света и свойствами материала поверхности
по курсу: Компьютерная графика

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4136

подпись, дата

Бобрович Н. С.

инициалы, фамилия

Санкт-Петербург 2022

Задание:

Включить источник света, задать отражающие свойства поверхностей: диффузное, зеркальное отражение, цвет поверхности. Вывести несколько объемных объектов. Каждый из них должен иметь различные свойства поверхностей (доминирующее зеркальное или диффузное отражение, различный цвет внешних поверхностей).

Листинг программы:

```

1 //Включить источник света, задать отражающие свойства поверхностей :
2 //диффузное, зеркальное отражение, цвет поверхности.
3 //Вывести несколько объемных объектов.Каждый из них должен иметь
4 //различные свойства поверхностей(доминирующее зеркальное или
5 //диффузное отражение, различный цвет внешних поверхностей).
6
7 #include <glut.h>
8 #include <iostream>
9 int spin = 0;
10 //Инициализация
11 void init(void)
12 {
13     glClearColor(0.0, 0.0, 0.0, 0.0);
14     glShadeModel(GL_SMOOTH);
15     glEnable(GL_LIGHTING); //включаем свет
16     glEnable(GL_LIGHT1); //включаем источник свет 1
17     glEnable(GL_LIGHT0); //включаем источник свет 0
18     glEnable(GL_DEPTH_TEST);
19 }
20
21
22 void drawSnowMan() { // рисуем снеговика
23
24     glColor3f(0.0f, 1.0f, 0.3f);
25     GLfloat position[] = { 0.0, 0.0, 1.5, 1.0 }; //направление источника
26     GLfloat white_light[] = { 0.8, 0.8, 0.8, 1.0 }; //цвет рассеянного
27     GLfloat mat_specular[] = { 1.0, 1.0, 1.0, 1.0 }; //цвет зеркального
28     GLfloat mat_shininess[] = { 100.0 };
29     GLfloat light_ambient[] = { 0.0, 0.4, 1.7, 1.0 }; //цвет фонового и
30
31     glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT); // очищаем
32     glLoadIdentity();
33     glTranslatef(0.0, 0.0, -5.0);
34
35     glLightfv(GL_LIGHT0, GL_POSITION, light_pos1); // задаем позицию ис
36     glLightfv(GL_LIGHT1, GL_POSITION, light_pos2); // задаем позицию и
37
38     glPushMatrix();
39     glTranslated(-1.7, 0.0, 0.0);
40     glLightfv(GL_LIGHT0, GL_DIFFUSE, white_light);
41     glutSolidTeapot(0.5); // чайник
42     glPopMatrix();
43
44     glPushMatrix();
45     glTranslatef(1.6, -1.0, 0.0);
46     glLightfv(GL_LIGHT0, GL_AMBIENT, light_ambient); // задаем источник
47     glLightfv(GL_LIGHT1, GL_AMBIENT, light_ambient); // задаем источник
48     drawSnowMan(); // снеговик

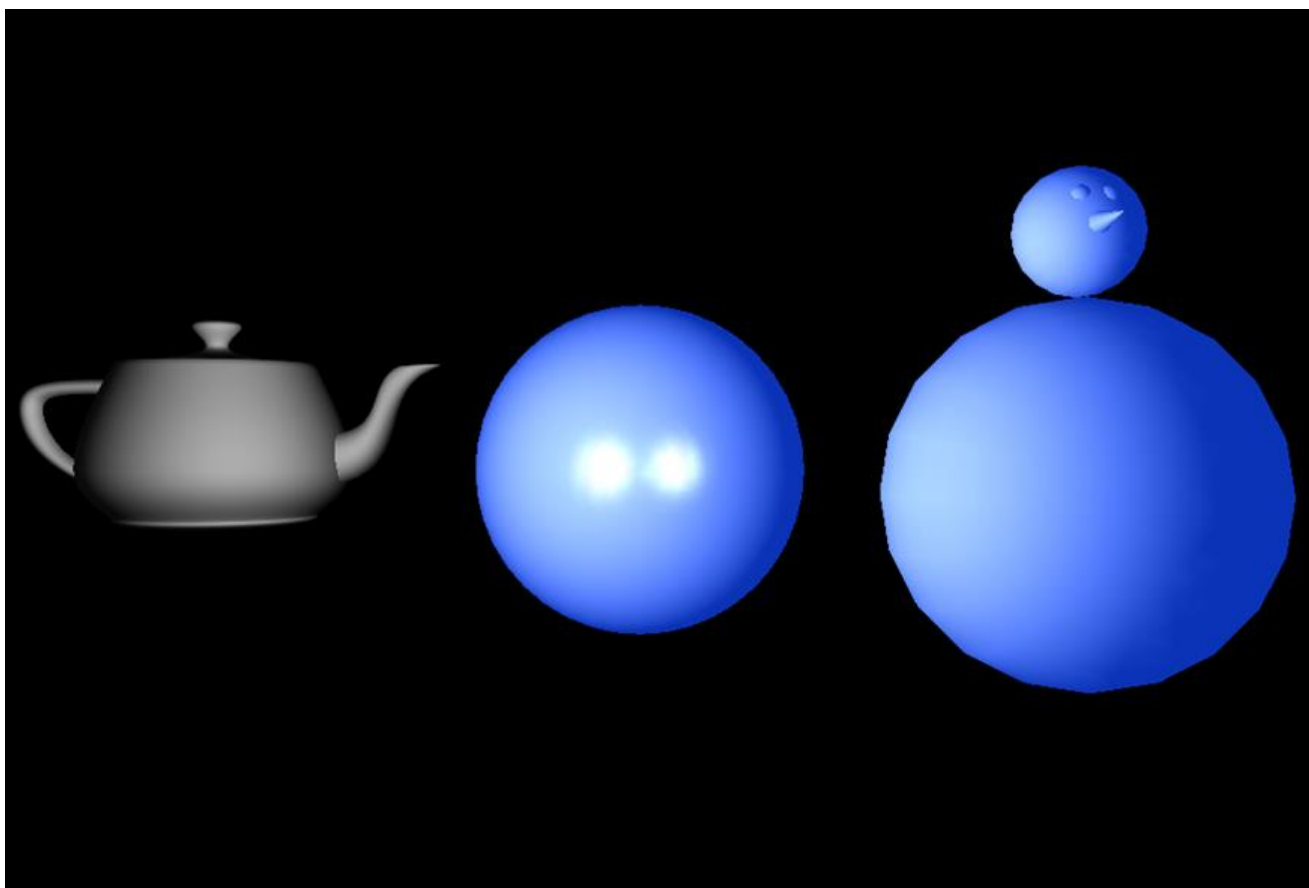
```

```

73     glPopMatrix();
74
75     glPushMatrix();
76     glTranslatef(-0.1, -0.2, -1.4);
77     glLightfv(GL_LIGHT0, GL_SPECULAR, white_light);
78     glLightfv(GL_LIGHT1, GL_SPECULAR, white_light);
79     glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, mat_specular); // задае
80     glMaterialfv(GL_FRONT_AND_BACK, GL_SHININESS, mat_shininess); // за
81     glutSolidSphere(0.8, 40, 40); // сфера
82     glPopMatrix();
83
84     glFlush();
85     glutSwapBuffers(); //смена буферов
86 }
87 //Изменение размеров окна
88 void reshape(int w, int h)
89 {
90     glViewport(0, 0, (GLsizei)w, (GLsizei)h);
91     glMatrixMode(GL_PROJECTION);
92     glLoadIdentity();
93     gluPerspective(40.0, (GLfloat)w / (GLfloat)h, 1.0, 20.0);
94     glMatrixMode(GL_MODELVIEW);
95     glLoadIdentity();
96 }
97
98 int main(int argc, char** argv)
99 {
100     glutInit(&argc, argv);
101     glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
102     glutInitWindowSize(1000, 700);
103     glutInitWindowPosition(100, 100);
104     glutCreateWindow("Лаба3");
105     init();
106
107     glutDisplayFunc(display); //отрисовка сцены
108     glutReshapeFunc(reshape);
109     glutMainLoop();
110     return 0;
111 }

```

Результат работы:



Выводы:

В результате выполнения работы были получены навыки работы с источником света и свойствами материала поверхности.