МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий

Кафедра технологий программирования

**ЛАБОРАТОРНАЯ РАБОТА №1**

по дисциплине: **«Алгоритмы компьютерной графики»**

на тему: **««Введение в программирование с использованием OpenGL»»**

ВЫПОЛНИЛ студент группы 16-ИТ-3

Яблонский А.С.

ПРОВЕРИЛ Глухов Д.О.

Полоцк, 2019 г.

**Цель работы**:

Ознакомиться с основами использования библиотеки OpenGL.

**Ход работы**

Необходимо:

1. Реализовать оконное приложение используя OpenGL API согласно примеру.
2. Изменить исходный код приложения для отображения примитива согласно индивидуальному варианту.

**Результаты выполнения лабораторной работы**

Листинг примера представлен ниже.

**Листинг 1.** Листинг примера

1. #include <stdlib.h>
2. #include <stdio.h>
3. #include <glut.h>
4. #include <cctype>
5. #define ESCAPE '\033'
6. using namespace std;
7. void printExtList(const char\* extensions) {
8. char name[1024];
9. int i = 0, j = 0;
10. printf("Supported extensions:\n");
11. for (i=0,j=0; extensions[i] != '\0'; i++) {
12. if (!isspace(extensions[i]))
13. name[j++] = extensions[i];
14. else {
15. name[j] = '\0';
16. printf("\t%s\n", name);
17. j = 0;
18. }
19. }
20. if (j > 0) {
21. name[j] = '\0';
22. printf("\t%s\n", name);
23. }
24. }
25. void init() {
26. glClearColor(1.0, 1.0, 1.0, 1.0);
27. glMatrixMode(GL\_PROJECTION);
28. glLoadIdentity();
29. glOrtho(0.0, 1.0, 0.0, 1.0, 0.0, 0.0);
30. }
31. void display() {
32. glClear(GL\_COLOR\_BUFFER\_BIT);
33. glBegin(GL\_POLYGON);
34. glColor3f(1.0, 0.0, 0.0);
35. glVertex3f(0.25, 0.25, 0.0);
36. glVertex3f(0.75, 0.25, 0.0);
37. glColor3f (0.0, 1.0, 0.0);
38. glVertex3f(0.75, 0.75, 0.0);
39. glVertex3f(0.25, 0.75, 0.0);
40. glEnd();
41. glFlush();
42. }
43. void reshape(GLint w, GLint h) {
44. glViewport(0, 0, w, h);
45. glMatrixMode(GL\_PROJECTION);
46. glLoadIdentity();
47. gluOrtho2D(0.0, 1.0, 0.0, 1.0);
48. }
49. void keyboard(unsigned char key, int x, int y) {
50. if (key == ESCAPE) exit(0);
51. }
52. int main(int argc, char\*\* argv) {
53. glutInit(&argc, argv);
54. glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);
55. glutInitWindowSize(250, 250);
56. glutInitWindowPosition(100, 100);
57. glutCreateWindow("Hello world!");
58. const char\* vendor = (const char\*)glGetString(GL\_VENDOR);
59. const char\* render = (const char\*)glGetString(GL\_RENDERER);
60. const char\* version = (const char\*)glGetString(GL\_VERSION);
61. const char\* extensions = (const char\*)glGetString(GL\_EXTENSIONS);
62. //printf(" Vendor: %s\n Render: %s\n Version %s\n ", vendor, render, version);
63. //printExtList(extensions);
64. init();
65. glutDisplayFunc(display);
66. glutReshapeFunc(reshape);
67. glutKeyboardFunc(keyboard);
68. glutMainLoop();
69. return 0;
70. }

Результат выполнения примера представлен ниже на рисунке 1.

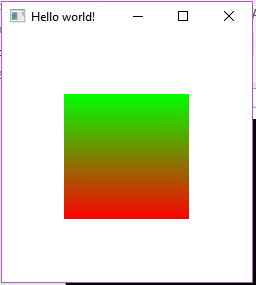
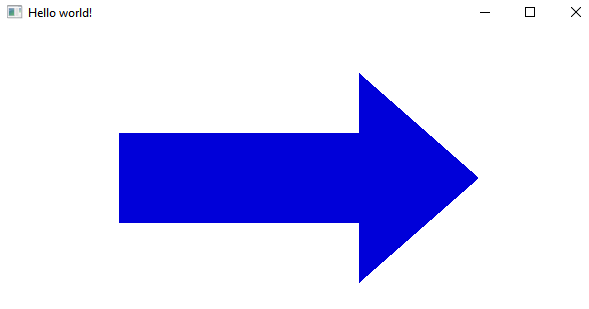


Рисунок 1 – Результат выполнения примера.

**Листинг 2.** Листинг функции display задания

1. void display() {
2. glClear(GL\_COLOR\_BUFFER\_BIT);
3. glBegin(GL\_POLYGON);
4. glColor3f(0.0, 0.0, 0.85);
5. glVertex2f(0.2, 0.65);
6. glVertex2f(0.6, 0.65);
7. glVertex2f(0.6, 0.35);
8. glVertex2f(0.2, 0.35);
9. glEnd();
10. glBegin(GL\_POLYGON);
11. glColor3f(0.0, 0.0, 0.85);
12. glVertex2f(0.6, 0.85);
13. glVertex2f(0.8, 0.5);
14. glVertex2f(0.6, 0.15);
15. glEnd();
16. glFlush();
17. }

Рисунок 2 – результат выполнения задания.

**Вывод**: в результате выполнения лабораторной работы было создано простое оконное приложение с использованием технологии OpenGL. В ходе выполнения работы были получены знания для работы с OpenGL.