

Задание

Нарисовать цветной треугольник используя старую версию OpenGL 2.0 .

Реализация

Открываем пустой проект, создаем .cpp файл и можем приступить к выполнению следующих шагов.

```
// Включаем стандартный заголовок
#include <GL/freeglut.h>
```

Дальше создаем функцию `main()` .

```
int main(int argc, char **argv){
```

В теле функции первым делом инициализируем `glut` :

```
    glutInit(&argc, argv);
```

Задаем режимы отображения для окна: `GLUT_DOUBLE`, к примеру, включает двойную буферизацию, буфер цвета `GLUT_RGBA` отвечает за цвет нашего объекта и `GLUT_DEPTH` за глубину (для определения переднего плана). Также задаем позицию окна на экране и размер окна (пиксель x пиксель). Теперь можно создавать OpenGL окно.

```
    glutInitDisplayMode(GLUT_DEPTH | GLUT_DOUBLE | GLUT_RGBA);
    glutInitWindowPosition(400, 100);
    glutInitWindowSize(500, 500);
    glutCreateWindow("OpenGL");
```

Нам необходимо добавить следующие функции обратного вызова: за рисование в окне отвечает команда `glutDisplayFunc()` , `glutMainLoop()` открывает определенный цикл, который ждет событие от оконной системы.

```
    glutDisplayFunc(function);
    glutMainLoop();
}
```

Теперь попробуем начать рисование в новой функции `function` .

```
void function(void) {  
    glBegin(GL_TRIANGLES);  
    glVertex3f(0.0, 0.7, 0.0);  
    glColor3f(1, 0, 1);  
    glVertex3f(-0.7, -0.7, 0.0);  
    glColor3f(1, 1, 0);  
    glVertex3f(0.7, -0.7, 0.0);  
    glColor3f(1, 1, 1);  
    glEnd();  
  
    glutSwapBuffers();  
}
```

Как можно увидеть, для прорисовки одной фигуры необходимо указать начало и конец обрабатываемых точек. И даже можно задать определенный цвет каждой точке. При запуске этого кода, у вас должен получиться такой треугольник:

Результаты

В результате мы получаем следующее изображение.



