

# ЗАНЯТИЕ 5

## ФУНКЦИИ ДЛЯ РИСОВАНИЯ

```
#include <iostream>
using namespace std;
void fun()          //описание функции
{
//тело функции состоит из одного оператора,он
выводит на экран звездочки и пробелы
cout << "* ";
}
int main ()
{
int i, j;
for (i=0; i<5; i++)
{
fun();          //вызов функции
cout<<"\n";    //перенос строки
}
system ("pause");
return 0;
}
```

# СТРУКТУРА ПРОГРАММЫ

```
void Display()
```

```
{
```

```
/*ТУТ РИСУЕМ*/
```

```
}
```

```
void Initialize() ()
```

```
{
```

```
/* ИНИЦИАЛИЗИРУЕМ ОКНО*/
```

```
}
```

```
int main()
```

```
{
```

```
/*Тут запускаем все созданные выше  
функции в бесконечном цикле*/
```

```
}
```

```
int main(int argc, char ** argv)
/* Это надо для обработки параметров командной строки
*/
/* Инициализируем GLUT */
glutInit(&argc, argv); //Установить начальные характеристики
окна
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);/*
Инициализируем режим прорисовки, GLUT_SINGLE означает,
что рисовать будем статическую картинку, а GLUT_RGB -
то, что цвета соответствуют RGB. */
glutInitWindowSize(400, 400);
glutInitWindowPosition(100, 200);
glutCreateWindow("Our first GLUT application!");
/* Обозначение размера, позиции и заголовка окна
соответственно */
glutDisplayFunc(Display);
Initialize();
glutMainLoop();
return 0; /* Обозначаем функцию прорисовки, вызываем
Initialize и запускаем цикл соот-но */
```

```
void Initialize() {  
    glClearColor(0.2, 1.0, 1.0, 1.0);  
    // ЗАДАЕМ ЦВЕТ ЗАЛИВКИ  
    glMatrixMode(GL_PROJECTION);  
    //устанавливаем проекцию, команды относятся проекты  
    glLoadIdentity();  
    gluOrtho2D(-200,200,-200,200);  
}  
/* Функция Initialize. Обозначаем цвет фона, выбираем  
тип  
матрицы, затем загружаемся с этими настройками.  
Потом устанавливаем диапазон изменения координат  
*/
```

```
void Display() {  
    glClear(GL_COLOR_BUFFER_BIT);  
    // очистить экран от любого цвета
```

ТУТ РИСУЕМ!!!!!!!

```
    glFlush();  
}
```

# ТОЧКИ

```
glBegin(GL_POINTS);  
    glVertex2f(10,20);  
    glVertex2f(30,40);  
glEnd();
```

# ЛИНИИ

```
glBegin(GL_LINES);  
    glVertex2f(10,20);  
    glVertex2f(30,40);  
glEnd();
```



# ПОЛИГОНЫ

```
glBegin(GL_POLYGON);  
    glVertex2f(10,20);  
    glVertex2f(30,40);  
glEnd();
```

# РАЗНОЕ

**glBegin(GL\_LINES);**

**glBegin(GL\_LINE\_STRIP);**

**glBegin(GL\_LINE\_LOOP);**

**glBegin(GL\_POLYGON);**

**glBegin(GL\_TRIANGLES);**

**glBegin(GL\_TRIANGLE\_STRIP);**