

Лабораторная №2. Многофайловая компоновка. Тестирование

Темы, рассматриваемые в лабораторной работе:

Научиться собирать проект из нескольких `cpp` - файлов. Формировать заголовочные файлы. Использовать директивы препроцессора. Ознакомиться со структурой проекта в Visual Studio. Сравнить результаты при компиляции `Debug` и `Release`. Научиться запускать готовый `exe`-файл из операционной системы. Научиться формировать проект для отправления его на проверку. Научиться подключать в решение проект с `unit`-тестами.

Задания:

1. Описать функции для решения задач. Создать проект, содержащий заголовочный файл, файл с реализацией функций и файл с функцией `main`.
 - `Distance` – возвращает расстояние между точками с вещественными координатами `x_1, y_1` и `x_2, y_2`
 - `Square` – возвращает площадь треугольника, заданного координатами вершин `x_a, y_a, x_b, y_b, x_c, y_c`
 - `Distance_to_line` - возвращает расстояние от точки, заданной координатами `x_a, y_a`, до прямой, заданной координатами двух точек, через которые она проходит `x_1, y_1` и `x_2, y_2`.
2. Научиться создавать тестовый проект, так как это описано в инструкции «Создание тестов».
3. Создать тестовый проект для набора функций `Distance`, `Square`, `Distance_to_line`. Написать набор тестов для каждой из реализованных функций.
4. Разработать наборы тестов для нескольких функций из лабораторной работы 1.

Дополнительные задания:

5. Функция `Check_triangle_type` возвращает тип треугольника (равносторонний, равнобедренный, произвольный), заданного координатами своих вершин. Тип функции может быть числовым (2- равносторонний, 1- равнобедренный, 0- произвольный).
6. Для классификации треугольников описать специальный тип. Описание помещается в заголовочном файле.

```
enum triangle_type {  
    tArbitrary = 0,      // произвольный  
    tIsoscel,           // равнобедренный  
    tEqui               // равносторонний  
};
```

Тогда тип результата новой функции `Check_triangle_type_new` – это специальный тип `triangle_type`. Почему в этом случае нельзя использовать перегрузку имени функции?

7. Описать структуру. Описание помещается в заголовочном файле.

```
struct point {  
    double x, y;  
};
```

Тогда параметры перегруженной функции `Check_triangle_type_new` будут иметь тип `point`, а обращение к координатам точки с именем `a` выглядит как `a.x` и `a.y`.

8. Одну из разработанных в задании 1 функции перегрузить, используя параметры типа `point`.