

**Министр науки и высшего образования Российской
Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

**«Национальный исследовательский университет
ИТМО»**

**Факультет информационных технологий и
программирования**

Лабораторная работа № 11

Использование заголовочных файлов.

**Выполнил студент группы № М3119
Черных Арсений Игоревич
Подпись:**

Проверил:
Повышев Владислав Вячеславович

Санкт-Петербург
2023

Текст задания

Задание

Разработать структуру, описывающую геометрическую фигуру на плоскости с помощью её координат, а также функции расчёта площади этой геометрической фигуры и её периметра (длины окружности). Дополнительно разработать функцию, позволяющую создавать структуру моделируемой геометрической фигуры на основании данных, введённых с консоли (самостоятельно определить формат этих вводимых данных). Вид геометрической фигуры и набор её рассчитываемых параметров определяются по номеру варианта. Разместить программный код функций, рассчитывающих параметры фигуры в отдельном файле исходного текста. Создать заголовочный файл, позволяющий вызывать функции созданного файла исходного текста из других файлов. Разместить в созданном заголовочном файле структуру, описывающую моделируемую геометрическую фигуру. Предусмотреть в созданном заголовочном файле защиту от повторного включения с помощью Include Guard. Создать файл исходного текста для размещения в нём точки входа. Разработать программный код функции main, создающий экземпляр структуры моделируемой фигуры, а также осуществляющий вызовы созданных функций расчёта параметров этой фигуры. Входные данные для моделирования фигуры вводить с консоли, результаты расчёта параметров фигуры также выводить на консоль.

Варианты геометрических фигур

1

Прямоугольник.

2

Квадрат.

3

Окружность (круг).

4

Параллелограмм.

5

Ромб.

6

Произвольный треугольник.

7

Правильный выпуклый пятиугольник.

8

Прямоугольный треугольник.

9

Равнобедренный треугольник.

10 Кольцо (фигура, созданная пересечением двух концентрических кругов).

№ варианта	Геометрическая фигура	Расчёт периметра/длины	Расчёт площади	Создание структуры по данным с консоли
1	1	×	×	
2	2	×	×	×
3	3	×	×	×
4	4	×	×	
5	5	×	×	
6	6	×	×	
7	7	×		×
8	8	×	×	×
9	9	×	×	
10	10	×	×	

№ варианта	Геометрическая фигура	Расчёт периметра/длины	Расчёт площади	Создание структуры по данным с консоли
11	1	×	×	
12	2	×	×	×
13	3	×	×	×
14	4	×	×	

Решение с комментариями

Работал с фигурой – параллелограмм. Заголовочный файл “header.h”. Макрос назвал GEOMETRY_H.

```
// main.c
```

```
#include <stdio.h>
```

```
#include "header.h"
```

```
int main() {
```

```
    struct Parallelogram parallelogram = create();
```

```
    double area_result = area(parallelogram);
```

```
    double length_result = length(parallelogram);
```

```
    printf("Area: %.1f\n", area_result);
```

```
    printf("Length: %.4f\n", length_result);
```

```
    return 0;
```

```
}
```

Header.h

```
#ifndef GEOMETRY_H
```

```
#define GEOMETRY_H
```

```
struct Parallelogram{
```

```
    double x1, y1;
```

```
    double x2, y2;
```

```
    double x3, y3;
```

```
};
```

```
double area(struct Parallelogram parallelogram);
```

```
double length(struct Parallelogram parallelogram);
```

```
struct Parallelogram create();
```

```
#endif
```

Geometry.c

```
#include <stdio.h>
```

```
#include <math.h>
```

```
#include "header.h"
```

```
double area(struct Parallelogram parallelogram) {  
    double vector1X = parallelogram.x2 - parallelogram.x1;  
    double vector1Y = parallelogram.y2 - parallelogram.y1;  
    double vector2X = parallelogram.x3 - parallelogram.x1;  
    double vector2Y = parallelogram.y3 - parallelogram.y1;  
    return fabs(vector1X * vector2Y - vector1Y * vector2X);  
}
```

```
double length(struct Parallelogram parallelogram) {  
    double side1 = sqrt(pow(parallelogram.x2 - parallelogram.x1, 2) +  
        pow(parallelogram.y2 - parallelogram.y1, 2));  
    double side2 = sqrt(pow(parallelogram.x3 - parallelogram.x2, 2) +  
        pow(parallelogram.y3 - parallelogram.y2, 2));  
    double side3 = sqrt(pow(parallelogram.x1 - parallelogram.x3, 2) +  
        pow(parallelogram.y1 - parallelogram.y3, 2));  
    return side1 + side2 + side3;  
}
```

```
struct Parallelogram create() {  
    struct Parallelogram parallelogram;  
    scanf("%lf %lf", &parallelogram.x1, &parallelogram.y1);  
    scanf("%lf %lf", &parallelogram.x2, &parallelogram.y2);  
    scanf("%lf %lf", &parallelogram.x3, &parallelogram.y3);  
    return parallelogram;  
}
```