

**Министерство 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

Выполнила студентка группы № М3118

Маркозубова Анастасия Кирилловна

Подпись:

Проверил:

Повышев Владислав Вячеславович

Санкт-Петербург
2023

Текст задания

Вариант №18. Прямоугольный треугольник

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.

Решение с комментариями

```
// main.c
#include <stdio.h>
#include "Triangle.h"

int main() {
    printf("Enter coordinates:");

    float x1, y1, x2, y2, x3, y3;

    // Ввод координат с консоли
    scanf("%f %f %f %f %f %f", &x1, &y1, &x2, &y2, &x3, &y3);

    // Создание структуры треугольника с учетом координат
    struct triangle triangle = makeTriangle(x1, y1, x2, y2, x3, y3);

    // Вывод площади и периметра треугольника на консоль
    printf("Area: %f\n", area(triangle));
    printf("Perimeter: %f\n", perimetr(triangle));

    return 0;
}

// triangle.h
// Защита от повторного включения с помощью Include Guard
#ifndef TRIANGLE_H
#define TRIANGLE_H
#include <stdio.h>
#include <math.h>

// Описание структуры точки на координатной плоскости
struct point {
    float x;
    float y;
};

// Описание структуры треугольника включающая в себя три точки треугольника
struct triangle {
    struct point a;
    struct point b;
    struct point c;
};

// Прототипы функций, описанных в function.c
struct triangle makeTriangle(float x1, float y1, float x2, float y2, float x3,
float y3);
float area(struct triangle triangle);
float perimetr(struct triangle triangle);

#endif
```

```

// function.c
#include "Triangle.h"

// Функция для расчета периметра прямоугольного треугольника
float perimetr(struct triangle triangle) {
    float side1 = sqrt(pow(triangle.a.x - triangle.b.x, 2) + pow(triangle.a.y -
triangle.b.y, 2));
    float side2 = sqrt(pow(triangle.b.x - triangle.c.x, 2) + pow(triangle.b.y -
triangle.c.y, 2));
    float side3 = sqrt(pow(triangle.c.x - triangle.a.x, 2) + pow(triangle.c.y -
triangle.a.y, 2));
    float perimeter = side1 + side2 + side3;
    return perimeter;
}

// Функция для расчета площади прямоугольного треугольника
float area(struct triangle triangle) {
    float triangleArea = 0.5 * fabs(triangle.a.x * (triangle.b.y - triangle.c.y) +
triangle.b.x * (triangle.c.y - triangle.a.y) + triangle.c.x * (triangle.a.y -
triangle.b.y));
    return triangleArea;
}

// Создание структуры треугольника
struct triangle makeTriangle(float x1, float y1, float x2, float y2, float x3,
float y3) {
    struct triangle newTriangle;
    newTriangle.a.x = x1; newTriangle.a.y = y1;
    newTriangle.b.x = x2; newTriangle.b.y = y2;
    newTriangle.c.x = x3; newTriangle.c.y = y3;
    return newTriangle;
}

```

```

Enter coordinates:2 2 2 8 6 2
Area: 12.000000
Perimeter: 17.211102

```