**Министр науки и высшего образования Российской̆ Федерации**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«Национальный исследовательский университет ИТМО»**

Факультет информационных технологий и программирования

Лабораторная работа №6

*Название работы:* Указатели и динамическая память.

**Выполнила студентка группы № M3117**

Виноградов Дмитрий Евгеньевич

**Подпись:**

**Проверил:**

Повышев Владислав Вячеславович

Задача:

Лабораторная работа 11. Использование заголовочных файлов

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 текса 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.

Решение:

Файл 11\_c.c:

Определение далее используемых функций

#include "11\_h.h"  
#include <math.h>  
  
float per(struct cord cord1,struct cord cord2, struct cord cord3, struct cord cord4){  
 return 2\*(abs(cord1.x-cord4.x)+sqrt(pow(abs(cord2.y-cord1.y),2)+pow(abs(cord2.x-cord1.x),2)));  
}//вычисление периметра по формуле  
  
float square(struct cord cord1,struct cord cord2, struct cord cord3, struct cord cord4){  
 return abs(cord2.y-cord1.y)\*abs(cord4.x-cord1.x);  
}//вычисление площади по формуле

Файл 11\_h.h:

Инициализация функций и структуры точек координат

#ifndef LAB11\_11\_H\_H  
#define LAB11\_11\_H\_H  
  
struct cord{  
 float x;  
 float y;  
};  
  
float per(struct cord,struct cord,struct cord,struct cord); //функция периметра  
float square(struct cord,struct cord,struct cord,struct cord); //функция площади  
  
#endif

Файл main.c:

#include <stdio.h>  
#include "11\_h.h"  
  
int main(){  
 struct cord cord1,cord2,cord3,cord4;  
 printf("input 4 dots of parallelogram in order 1 to 4");  
 scanf("%f%f%f%f%f%f%f%f",&cord1.x,&cord1.y,&cord2.x,&cord2.y,&cord3.x,&cord3.y,&cord4.x,&cord4.y); //ввод 8 координат 4 точек   
  
 printf("%f\n", per(cord1,cord2,cord3,cord4)); //вывод периметра паралеллограмма  
 printf("%f", square(cord1,cord2,cord3,cord3));  
}//вывод площади паралеллограмма