Министерство высшего образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет» (ПНИПУ)**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

по лабораторной работе №7

Тема: «Перегрузка функций и функции с переменным числом параметров»

Выполнил

Студент группы РИС-22-2б

Будин Д.В.

Проверил доц. Кафедры ИТАС

Полякова Ольга Андреевна

Пермь 2022

**7.2. Перегрузка функций.**

Решить указанную в варианте задачу, используя функции с переменным число параметров.

24. Написать функцию (или макроопределение), которая находит длину стороны по координатам его точек. Написать функцию square, которая вычисляет площадь треугольника, заданного координатами вершин. Написать функцию squaren c переменным числом параметров, которая определяет площадь выпуклого многоугольника, заданного координатами своих вершин.

**Анализ**

Чтобы вычислить площадь выпуклого многоугольника, можно сложить все входящие в него треугольники.

1. Функция side\_length – вычисление длины отрезка по координатам:

return sqrt(pow((y2 - y1), 2) + pow((x2 - x1), 2));

1. Функция triangle – вычисление площади треугольника:

double AB, BC, CD;

AB = side\_length(x1, y1, x2, y2) – длина отрезка AB(так же и с BC, CA)

double p = (AB + BC + CD) / 2 – полупериметр

return sqrt(p \* (p - AB) \* (p - BC) \* (p - CD)) – площадь по формуле Герона

1. double squaren(int k, ...) – ф-ия с переменным число параметров, вычисляющая площадь многоугольника.
2. va\_list args;

va\_start(args, k) – создание указателя k на первый аргумент.

5. x1 = va\_arg(args, int);

y1 = va\_arg(args, int); -

x2 = va\_arg(args, int);

…

1. total = triangle(x1, y1, x2, y2, x3, y3) – вычисление первого треугольника вписанного в многоугольник.
2. Цикл по всем оставшимся треугольникам:

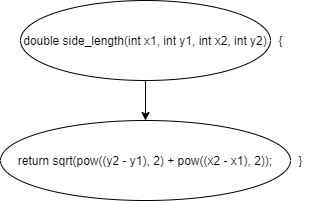
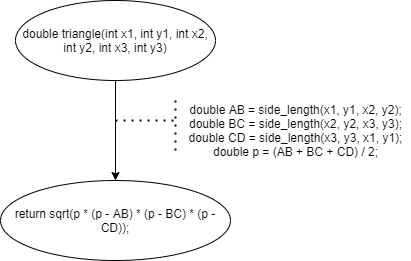
for (int i = 0; i < k / 2 - 3; i++)

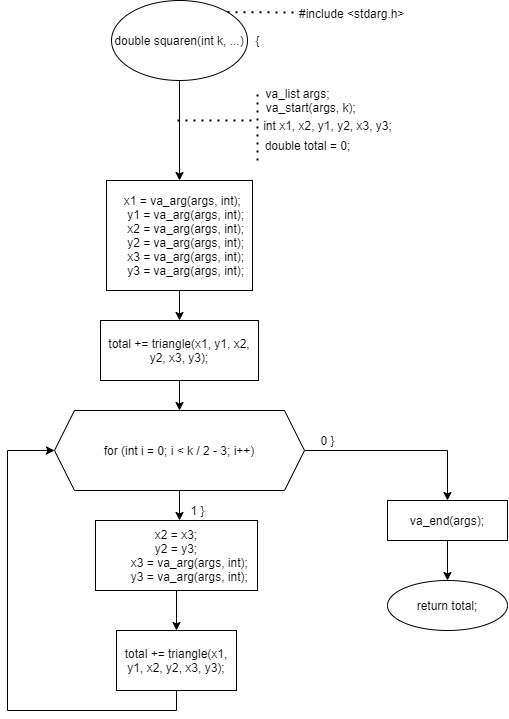
x2 = x3; y2 = y3;

x3 = va\_arg(args, int); y3 = va\_arg(args, int);

total += triangle(x1, y1, x2, y2, x3, y3);

1. va\_end(args) – окончание аргументов и вывод результата: return total;

**Алгоритм программы**

**Код программы**

#include <iostream>

#include <cmath>

#include <stdarg.h>

using namespace std;

double side\_length(int x1, int y1, int x2, int y2)

{

return sqrt(pow((y2 - y1), 2) + pow((x2 - x1), 2));

}

double triangle(int x1, int y1, int x2, int y2, int x3, int y3)

{

double AB = side\_length(x1, y1, x2, y2);

double BC = side\_length(x2, y2, x3, y3);

double CD = side\_length(x3, y3, x1, y1);

double p = (AB + BC + CD) / 2;

return sqrt(p \* (p - AB) \* (p - BC) \* (p - CD));

}

double squaren(int k, ...)

{

va\_list args;

va\_start(args, k);

double total = 0;

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

x1 = va\_arg(args, int);

y1 = va\_arg(args, int);

x2 = va\_arg(args, int);

y2 = va\_arg(args, int);

x3 = va\_arg(args, int);

y3 = va\_arg(args, int);

total += triangle(x1, y1, x2, y2, x3, y3);

for (int i = 0; i < k / 2 - 3; i++)

{

x2 = x3;

y2 = y3;

x3 = va\_arg(args, int);

y3 = va\_arg(args, int);

total += triangle(x1, y1, x2, y2, x3, y3);

}

va\_end(args);

return total;

}

int main()

{

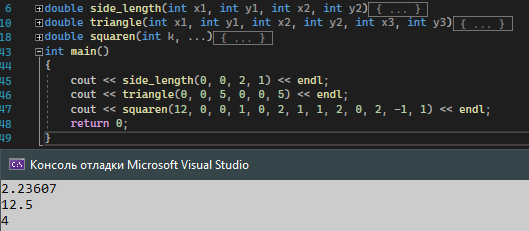
cout << side\_length(0, 0, 2, 1) << endl;

cout << triangle(0, 0, 5, 0, 0, 5) << endl;

cout << squaren(12, 0, 0, 1, 0, 2, 1, 1, 2, 0, 2, -1, 1) << endl;

return 0;

}

**Результаты работы:**