Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет», ПНИПУ

ЛАБОРАТОРНАЯ РАБОТА

ФУНКЦИИ С ПЕРЕМЕННЫМ ЧИСЛОМ ПАРАМЕТРОВ

Выполнил: студент группы РИС-23-3б

Артем Владимирович Швецов

Проверила: доцент кафедры ИТАС

Ольга Андреевна Полякова

Пермь 2024

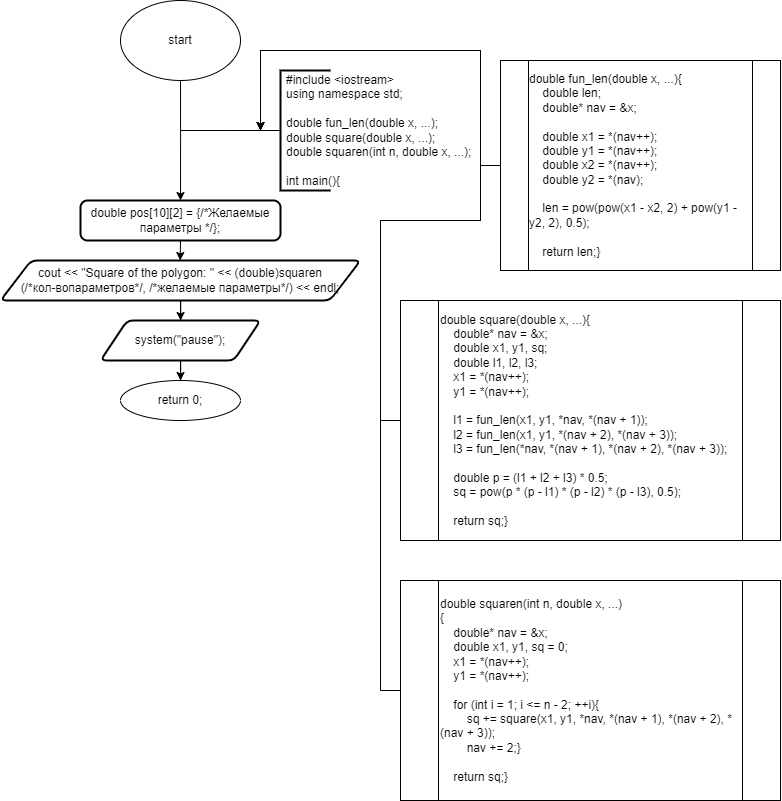
**Постановка задачи**

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 задачи**

* Функции square и squaren опираются на результаты предыдущих функций (длина стороны для square (нахождение площади через 3 стороны), square для squaren (нахождение площади многоугольника как суммы площадей треугольников)).
* Каждая функция возвращает запрашиваемую величину.
* Проход по параметрам осуществляется с помощью указателей.

Блок-схема



Код

#include <iostream>

using namespace std;

double fun\_len(double x, ...);

double square(double x, ...);

double squaren(int n, double x, ...);

int main()

{

double pos[10][2] = {

{1.0, 1.0},

{4.0, 3.0},

{4.0, 1.0},

{1.0, 3.0},

{0.0, 0.0},

{0.0, 0.0},

{0.0, 0.0},

{0.0, 0.0},

{0.0, 0.0},

{0.0, 0.0}};

cout << (double) squaren(4, pos[0][0], pos[0][1], pos[1][0], pos[1][1], pos[2][0], pos[2][1], pos[3][0], pos[3][1]) << endl;

system("pause");

return 0;

}

double fun\_len(double x, ...)

{

double len;

double\* nav = &x;

double x1 = \*(nav);

nav++;

double y1 = \*(nav);

nav++;

double x2 = \*(nav);

nav++;

double y2 = \*(nav);

len = pow(pow(x1 - x2, 2) + pow(y1 - y2, 2), 0.5);

return len;

}

double square(double x, ...)

{

double\* nav = &x;

double x1, y1, sq;

double l1, l2, l3;

x1 = \*(nav++);

y1 = \*(nav++);

l1 = fun\_len(x1, y1, \*nav, \*(nav + 1));

l2 = fun\_len(x1, y1, \*(nav + 2), \*(nav + 3));

l3 = fun\_len(\*nav, \*(nav + 1), \*(nav + 2), \*(nav + 3));

double p = (l1 + l2 + l3) \* 0.5;

sq = pow(p \* (p - l1) \* (p - l2) \* (p - l3), 0.5);

return sq;

}

double squaren(int n, double x, ...)

{

double\* nav = &x;

double x1, y1, sq = 0;

x1 = \*(nav++);

y1 = \*(nav++);

for (int i = 1; i <= n - 2; ++i)

{

sq += square(x1, y1, \*nav, \*(nav + 1), \*(nav + 2), \*(nav + 3));

nav += 2;

}

return (double)sq;

}

Решение



Выводы

В ходе данной работы были изучены функции с переменным числом параметров и некоторые особенности работы с ними.

Github

https://github.com/Hitikov/Lab\_7