

Отчет

по лабораторной работе 1444 “**Накормить** **элефпотама**”

по дисциплине «Алгоритмы и структуры данных»

Авторы:

Полит Алексей Денисович

Факультет: СУиР

Группа: R3235

Преподаватель: Тропченко Андрей Александрович



УНИВЕРСИТЕТ ИТМО

1. Задача

Гарри Поттер сдает экзамен по предмету «Уход за магическими существами». Его задание — накормить карликового элфполама. Гарри помнит, что элфполамы отличаются прямолинейностью и невозмутимостью. Они настолько прямолинейны, что ходят строго по прямой, и настолько невозмутимы, что заставить их идти можно, только если привлечь его внимание к чему-нибудь действительно вкусному. И главное, наткнувшись на цепочку своих собственных следов, элфполам впадает в ступор и отказывается идти куда-либо. По словам Хагрида, элфполамы обычно возвращаются домой, идя в обратную сторону по своим собственным следам. Поэтому они никогда не пересекают их, иначе могут заблудиться. Увидев свои следы, элфполам детально вспоминает все свои перемещения от выхода из дома (поэтому-то они и ходят только по прямой и лишней раз не меняют направление — так легче запоминать). По этой информации элфполам вычисляет, в какой стороне расположена его нора, после чего поворачивается и идет прямо к ней. Эти вычисления занимают у элфполама некоторое (довольно большое) время. А то, что некоторые невежды принимают за ступор, на самом деле есть проявление выдающихся вычислительных способностей этого чудесного, хотя и медленно соображающего животного!

Любимое лакомство элфполамов — слоновьи тыквы, именно они и растут на лужайке, где Гарри должен сдать экзамен. Перед началом испытания Хагрид притащит животное к одной из тыкв. Скармливая элфполаму очередную тыкву, Гарри может направить его в сторону любой оставшейся тыквы. Чтобы сдать экзамен, надо провести элфполама по лужайке так, чтобы тот съел как можно больше тыкв до того, как наткнется на свои следы.

2. Исходные данные

В первой строке входа находится число N ($3 \leq N \leq 30000$) — количество тыкв на лужайке. Тыквы пронумерованы от 1 до N , причем номер один присвоен той тыкве, у которой будет стоять элфполам в начале экзамена. В следующих N строках даны координаты всех тыкв по порядку. Все координаты — целые числа от -1000 до 1000 . Известно, что положения всех тыкв различны, и не существует прямой, проходящей сразу через все тыквы.

3. Код программы

```

#include <iostream>

#include <cmath>

using namespace std;

#define PI 3.14159265;

struct point {

    float length;

    double degrees;

    int index;

};

int compare_points(const void *var1, const void *var2) {

    const point*p1 = (point*) var1;

    const point*p2 = (point*) var2;

    if (p1->degrees - p2->degrees > 1e-10)

        return 1;

    else if (p1->degrees - p2->degrees < -1e-10)

        return -1;

    else

        if (p1->length > p2->length)return 1;

        else return -1;

}

int main() {

    int n;

    short f_x,f_y,x,y;

    cin >> n >> f_x >>f_y;

    point dots[n];

    dots[0].degrees = -1;

    dots[0].length = 0;

    dots[0].index = 1;

```

```

for (int i = 1; i < n; i++) {
    cin >> x;
    cin >> y;
    dots[i].index = i+1;
    dots[i].length = (x - f_x)*(x - f_x)+(y - f_y)*(y - f_y);
    dots[i].degrees = atan2(y - f_y, x - f_x) * 180.0 / PI;
    if (y - f_y < 0) dots[i].degrees += 360;
}
qsort(dots, n, sizeof(point), compare_points);
int start_point = 1;
double max_a = dots[1].degrees - dots[n-1].degrees+360 ;
for (int i = 1; i < n - 1; i++) {
    if (dots[i + 1].degrees - dots[i].degrees > max_a) {
        max_a = dots[i + 1].degrees - dots[i].degrees;
        start_point = i+1;
    }
}
cout << n << endl << 1 << endl;

for (int i = start_point; i < n; i++) {
    cout << dots[i].index << endl;
}
for (int i = 1; i < start_point; i++) {
    cout << dots[i].index << endl;
}
return 0;
}

```

4.

9368744	14:55:30 17 май 2021	Aleksey	1444. Накормить слеппотаму	G++ 9.2 x64	Accepted		0.109	1 120 КБ
---------	-------------------------	-------------------------	--	-------------	----------	--	-------	----------

Описание алгоритма:

Первую точку берём как начало координат. По ходу считывания записывается угол каждой точки относительно оси Ох и расстояние до нового начала координат. Сортируем точки по углу и по расстоянию до начала координат. Находим максимальный угол между направлениями на две соседние точки, чтобы найти начало обхода, исключаящее пересечение с маршрутом от 1 до 2 точки. После всех операций выводится обход точек "по кругу"