

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №2
по «Алгоритмам и структурам данных»

Выполнил:

Студент группы Р3210

Кадыров А.Х.

Преподаватели:

Косяков М.С.

Тараканов Д.С.

Санкт-Петербург

2019

Задача №1726 «Кто ходит в гости»

Пояснение к примененному алгоритму:

Люди ходят только параллельно оси ox и oy , то результат - это сумма расстояний между каждым домом с каждым по x и y координатам. Сохраним все координаты и отсортируем. Между каждой парой соседних домов узнаем расстояние g и узнаем какое количество человек по ней проходит (а именно i человек справа и $(n - i)$ слева).

Следовательно, суммарно по этой тропинке пройдут $g \cdot i \cdot (n - i)$. Итоговую сумму разделим на количество тропинок между домами для получения среднего результата

Код:

```
#include <iostream>
```

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
bool comp(long long a, long long b) { return a < b; }
```

```
void quick_sort(long long *vector, int first, int last) {
```

```
    long long mid;
```

```
    int f = first, l = last;
```

```
    mid = vector[(f + l) / 2];
```

```
    do {
```

```
        while (comp(vector[f], mid))
```

```
            f++;
```

```
        while (comp(mid, vector[l]))
```

```
            l--;
```

```
        if (f <= l) {
```

```
            swap(vector[f], vector[l]);
```

```
            f++;
```

```
            l--;
```

```
        }
```

```
    } while (f < l);
```

```
    if (first < l)
```

```
        quick_sort(vector, first, l);
```

```
    if (f < last)
        quick_sort(vector, f, last);
}
```

```
int main() {
    long long n;
    cin >> n;
    long long x_s[n];
    long long y_s[n];
    long long sum = 0;
    for (long long i = 0; i < n; ++i) {
        cin >> x_s[i] >> y_s[i];
    }
```

```
    quick_sort(x_s, 0, n - 1);
    quick_sort(y_s, 0, n - 1);
```

```
    for (long long i = 0; i < n - 1; ++i) {
        sum += (x_s[i + 1] - x_s[i]) * (i + 1) * (n - i - 1);
        sum += (y_s[i + 1] - y_s[i]) * (i + 1) * (n - i - 1);
    }
```

```
    sum = 2 * sum / (n * (n - 1));
    cout << sum;
    return 0;
}
```