

Czech ACM Student Chapter

Charles University in Prague
Slovak University of Technology
University of Žilina
Matej Bel University in Banská Bystrica

Czech Technical University in Prague

Technical University of Ostrava
Pavol Jozef Šafárik University in Košice
Masaryk University
University of West Bohemia



CTU Open Contest 2018

Moving Furniture

`furniture.c`, `furniture.cpp`, `furniture.c11`, `Furniture.java`, `furniture.py`

Binary Casino had been open nonstop for a long period of time. It desperately needed renovation as the substantial damage to its carpet and furniture did not look good. While renovation took place, four-legged game tables from all game rooms were deposited in a storage room.

Your task is not difficult. You were asked by your boss to move the game tables to the exactly same place where they were positioned before. Each game table desk is a square and the table legs are located exactly under the corners of the desk. A good thing is that there are small hollows made by the table legs in the carpet at places where the game tables were standing before renovation and you know how many tables were in the room. Another good thing is that your boss does not remember where exactly were the tables placed. The bad thing is that your boss knows the total area of all game tables in the room and insists on the number being preserved.

You have to use every hollow in the carpet to place the game tables, placing one table leg into each hole. The tables cannot overlap.

Input Specification

The first line of input contains an integer N ($1 \leq N \leq 3 \cdot 10^3$), the number of game tables. After that $4N$ lines follow, each containing two integers X and Y ($-10^9 \leq X, Y < 10^9$) which represent coordinates of a hollow in the carpet. The hollows are listed in an arbitrary order and no two hollows may have the same coordinates.

Output Specification

Output a single number – the sum of areas of all game tables in the room. The output will be considered correct if it differs by at most 10^{-3} from the correct answer.

Sample Input 1

```
4
1 0
1 -1
0 0
0 1
-2 -1
-2 1
-1 -2
-2 -3
-3 -2
-1 1
-1 -1
-3 -1
-3 1
0 -1
-2 3
0 3
```

Output for Sample Input 1

```
11
```

Sample Input 2

```
1
0 0
3 4
-1 7
-4 3
```

Output for Sample Input 2

```
25.0000
```