



## Task 1: Best Place

IOI 2020 will be held in Singapore, hosted by School of Computing of the National University of Singapore (NUS), supported by Singapore Ministry of Education (MOE) and Singapore Exhibition & Convention Bureau (SECB).

One of the challenges of hosting such a large event is to find a suitable location for the event. A common suggestion would be to select a location that reduces the travelling distance for the participants. You are thus tasked to write a program to find one such location.

Suppose there are  $N$  participants for IOI 2020. Assume the  $i^{\text{th}}$  participant's house is located on  $(X_i, Y_i)$  on the cartesian plane. You aim to find a location to hold IOI 2020 such that the total sum of travelling distance is minimized. Assume IOI 2020 is chosen to be at  $(X, Y)$ , the travelling distance for the  $i^{\text{th}}$  participant is the Manhattan distance between  $(X, Y)$  and  $(X_i, Y_i)$ , i.e.,  $|X - X_i| + |Y - Y_i|$ .

If there are multiple locations where the total sum of travelling distance is minimized, you are to output any one of these coordinates. It is possible for the location of the event be the same as the location of a participant's house. However, the event must be held at a position  $(X, Y)$  where  $X$  and  $Y$  are integers. It is also possible that more than 1 participant lives at the same coordinates. In that case, you are to sum their travelling distance separately. (Assume they will travel separately.)

### Input

Your program must read from standard input.

The input will start with a single integer,  $N$ , in a single line.  $N$  denotes the total number of participants.

$N$  lines will then follow with 2 integers each, the  $i^{\text{th}}$  line will contain  $X_i$  and  $Y_i$ . This indicates that the coordinate of the  $i^{\text{th}}$  participant's house is at  $(X_i, Y_i)$ .

### Output

Your program must output to standard output only.

Output 2 integers  $(X, Y)$  on the same line, space separated.  $(X, Y)$  will denote the coordinates of the event where by the total sum of travelling distance would be minimized.



## Subtasks

The maximum execution time on each instance is 1.0s. Your program will be tested on input instances that satisfy the following restrictions:

Subtask	Marks	$N$	$X_i, Y_i$
1	3	$N = 2$	$0 \leq X_i, Y_i \leq 10^9$
2	20	$2 \leq N \leq 1\,000$	$0 \leq X_i \leq 1\,000, Y_i = 0$
3	28	$2 \leq N \leq 100\,000$	$0 \leq X_i \leq 10^9, Y_i = 0$
4	13	$2 \leq N \leq 100$	$0 \leq X_i, Y_i \leq 100$
5	17	$2 \leq N \leq 1\,000$	$0 \leq X_i, Y_i \leq 10^9$
6	19	$2 \leq N \leq 100\,000$	$0 \leq X_i, Y_i \leq 10^9$

## Sample Testcase 1

This testcase is valid for all subtasks.

Input	Output
2 1 0 4 0	3 0

Note that this is not the only output that would be accepted. In particular, the following alternative outputs will also be accepted for Sample Testcase 1.

- 1 0
- 2 0
- 4 0

## Sample Testcase 1 Explanation

Regardless of whether we choose  $(1, 0)$ ,  $(2, 0)$ ,  $(3, 0)$  or  $(4, 0)$ , the sum of total travelling distance will still be 3.

## Sample Testcase 2

This testcase is only valid for subtasks 2 to 6.

Input	Output
6 1 0 3 0 5 0 7 0 9 0 11 0	7 0



Note that this is not the only output that would be accepted. In particular, the following alternative outputs will also be accepted for Sample Testcase 2.

- 5 0
- 6 0

### Sample Testcase 3

This testcase is only valid for subtasks 4 to 6.

Input	Output
9 1 16 3 12 5 6 7 10 9 8 11 4 13 14 15 2 17 18	9 10

Note that this is the only output that is accepted for Sample Testcase 3.