

# Diagonal sums

SEASON 6 – ROUND SIX – 100 points



One day Ivancho was digging through grandma's cellar and found an **NxN** table of numbers. With it he found an attached, yellowed paper with a tiny inscription:

*"Think of a cell from the table and follow the two diagonals, on which it lies. Find the sum of the numbers on the two diagonals and write it here. Then make a wish – if your calculations are correct, it will come true."*

Of course, Ivancho sees this as an opportunity to win Sashka's heart. He has other wishes though, so he intends to repeat the procedure **Q** times.

Your task is to help him. Given a number **Q** and **Q** pairs of indices of the chosen by Ivancho cells, you must print the sum of the numbers in the diagonals, which contain the given cell. The number in the given cell is added to the sum only once.

## Input

The first row of the file `diagonalsum.in` contains a positive integer **N**. **N** rows follow, each containing **N** space separated integers, describing the contents of the table.

On a new line is the integer **Q**. The last **Q** rows each contain a pair of cell indices. The indexing of the numbers is from **1 to N**. The first index is the row – from top to bottom, the second is the column – from left to right (standard indexing).

## Output

In the output file `diagonalsum.out` print **Q** rows. On each row print the respective diagonal sum.

## Constraints

$$1 \leq N \leq 10^3$$

$$1 \leq Q \leq 10^5$$

The values of the numbers in the table are between 1 and 1000.

**Time limit: 1.2 sec**

**Memory limit: 256 MB**

# Diagonal sums

SEASON 6 – ROUND SIX – 100 points



## Example testcase

Input (diagonalsum.in)	Output (diagonalsum.out)
6	133
1 16 7 9 36 6	85
17 26 2 15 20 27	204
3 14 34 24 4 19	133
25 35 22 31 12 30	142
8 18 5 29 13 10	
33 23 11 32 21 28	
5	
1 1	
1 3	
4 3	
6 6	
3 5	