Secondary breakeven positions   X58714_en

In an integer matrix with dimensions $f \times c$, a matrix position (i, j) is called *breakeven* if the sum of elements of the upper right submatrix and the sum of the elements of the lower left submatrix are equal. The upper right submatrix is determined by the positions $(0, j)$ and $(i, c - 1)$, while the lower left submatrix is determined by the positions $(i, 0)$ and $(f - 1, j)$.
For example, given the matrix:

```
8    5    3    0
6    4    1    9
0   -5    7   -1
```

position $(1, 2)$ is breakeven since both sums are equal: the sum of the upper right submatrix is $3 + 0 + 1 + 9 = 13$, while the sum of the lower left submatrix is $6 + 4 + 1 + 0 + (-5) + 7 = 13$. Given an integer matrix and a sequence of pairs (i, j), for each pair the program writes ''si'' and the sum of one of the submatrices if it is a breakeven position, or writes ''no'' and the sum of the two submatrices (the sum of the upper right submatrix first) otherwise, following the format of the examples.
In order to be valid, your program must use the following definition:

```
typedef vector<int> Fila;
typedef vector<Fila> Matriz;
```

and it must implement and use the following function:

```
// Pre: f > 0, c > 0; there is an integer matrix in the input channel
// Post: returns the integer matrix read from the input channel
Matriz read_matriz(int f, int c) {
    ...
}
```

Exam score: 2.50 **Automatic part:** 40.00%

Input

The input begins with two integers f, c greater than 0 followed by an integer matrix with dimensions $f \times c$. Finally, a sequence of pairs (i, j) where i represents valid row and j represents valid column of the matrix.

Output

For each pair (i, j) of the sequence, write ''si'' and the sum of one of the submatrices if it is a breakeven position, or write ''no'' and the sum of the two submatrices otherwise (the sum of the upper right submatrix first), following the format of the examples.

Sample input 1

```
3 4
```

```
8    5    3    0
6    4    1    9
0   -5    7   -1
```

```
1 2
0 0
2 3
0 3
2 0
```

Sample output 1

```
si: 13
no: 16, 14
no: 8, 1
no: 0, 37
no: 37, 0
```

Sample input 2

```
1 1
4

0 0
```

Sample output 2

```
si: 4
```

Problem information

Author : Pro1
Generation : 2021-01-12 15:23:36