Maximum Coins

Problem

Mike has a square matrix with **N** rows and **N** columns. Cell (i,j) denotes the cell present at row i and column j. Cell (1,1) denotes the top left corner of the matrix. Each cell has some amount of coins associated with it and Mike can collect them only if he visits that cell. $C_{i,j}$ represents the number of coins in cell with row i and column j. From a cell (i,j), Mike can decide to go to cell (i+1,j+1) or cell (i-1,j-1), as long as the cell lies within the boundaries of the matrix and has not been visited yet. He can choose to start the journey from any cell and choose to stop at any point. Mike wants to maximize the number of coins he can collect. Please help him determine the maximum number of coins he can collect.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case begins with a line containing the integer **N**. The next **N** lines contain **N** integers each. The j-th integer in the i-th line represents the number of coins $C_{i,j}$ in cell (i,j).

Output

For each test case, output one line containing `Case #x: y`, where `x` is the test case number (starting from 1) and `y` is the maximum number of coins Mike can collect.

Limits

Time limit: 20 seconds.
Memory limit: 1 GB.
$1 \le T \le 100$.
$0 \le C_{i,j} \le 10^7$.

Test Set 1

$1 \le N \le 100$.

Test Set 2

$1 \le N \le 10^3$ in at most 10 cases.
$1 \le N \le 100$ in all other cases.

Sample

Sample Input	Sample Output
2 3 1 2 5 3 6 1	Case #1: 14 Case #2: 9

```
12 2 7
5
0 0 0 0 0
1 1 1 1 0
2 2 2 8 0
1 1 1 0 0
0 0 0 0 0
```

In Sample Case #1, the maximum number of coins collected can be 14, if Mike follows this path:
(1,1) -> (2,2) -> (3,3)

In Sample Case #2, the maximum number of coins collected can be 9, if Mike follows this path:
(2,3) -> (3,4).