

Problem D. Deleting

Input file: *standard input*
Output file: *standard output*
Time limit: 4 seconds
Memory limit: 512 mebibytes

You are given an array $[1, 2, \dots, n]$, where the number of elements n is even.

In one operation, you can delete two adjacent elements of the array. If these elements are i and j , the cost of this operation is $cost(i, j)$.

In $\frac{n}{2}$ operations, all elements will be deleted. The cost of deleting the whole array is defined as the largest cost among all the $\frac{n}{2}$ operations.

What is the smallest possible cost of deleting the whole array?

Input

The first line of the input contains a single integer n ($2 \leq n \leq 4000$, n is even).

We are kind today. So we won't provide unnecessary input. It can be shown that it's impossible for two numbers of the same parity to be adjacent at any point, so we won't provide costs for those pairs.

The i -th of the next $n - 1$ lines contains $\lfloor \frac{n-i+1}{2} \rfloor$ integers. If i is even, these integers are $cost(i, i+1), cost(i, i+3), \dots, cost(i, n-1)$. Otherwise, they are $cost(i, i+1), cost(i, i+3), \dots, cost(i, n)$.

It is guaranteed that the costs form a permutation of numbers from 1 to $(\frac{n}{2})^2$.

Output

Output a single integer: the smallest possible cost of deleting the whole array.

Examples

standard input	standard output
2 1	1
6 2 1 3 4 5 6 7 8 9	6
10 20 21 2 11 25 3 24 18 8 6 17 7 5 22 4 23 14 15 1 19 16 12 10 13 9	14

Note

In the first example, the array is $[1, 2]$, and $cost(1, 2) = 1$. So, the only way to delete the array has the total cost of 1.



In the second example, one of the ways to delete the array is:

- $[1, 2, 3, 4, 5, 6] \rightarrow [1, 2, 5, 6]$, deleting pair $(3, 4)$ with cost 6.
- $[1, 2, 5, 6] \rightarrow [1, 6]$, deleting pair $(2, 5)$ with cost 5.
- And then deleting pair $(1, 6)$ with cost 3.

The total cost is therefore $\max(6, 5, 3) = 6$.