Problem H
Running Routes

The administrators at Polygonal School want to increase enrollment, but they are unsure if their gym can support having more students. Unlike a normal, boring, rectangular gym, the gym floor at Polygonal is a regular n-sided polygon! They affectionately refer to the polygon as P.

The coach has drawn several running paths on the floor of the gym. Each running path is a straight line segment connecting two distinct vertices of P. During gym class, the coach assigns each student a different running path, and the student then runs back and forth along their assigned path throughout the class period. The coach does not want students to collide, so each student's path must not intersect any other student's path. Two paths intersect if they share a common point (including an endpoint).

Given a description of the running paths in P, compute the maximum number of students that can run in gym class simultaneously.

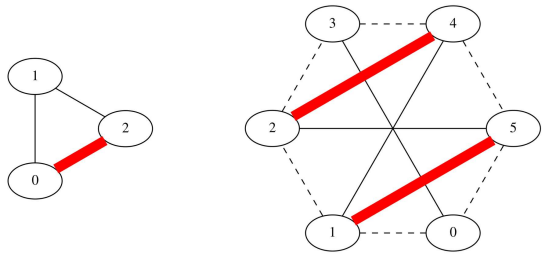


Figure H.1: Illustrations of the two sample inputs, with possible solutions highlighted in thick red lines. Solid black lines represent running paths that are not assigned to a student, and dashed black lines are used to show the boundary of P in places where no running path exists.

Input

The first line contains an integer n ($3 \leq n \leq 500$), the number of vertices in P. (The vertices are numbered in increasing order around P.) Then follows n lines of n integers each, representing a $n \times n$ symmetric binary matrix which we'll call M. The j^{th} integer on the i^{th} line M_{ij} is 1 if a running path exists between vertices i and j of the polygon, and 0 otherwise. It is guaranteed that for all $1 \leq i, j \leq n$, $M_{ij} = M_{ji}$ and $M_{ii} = 0$.

Output

Print the maximum number of students that can be assigned to run simultaneously on the running paths, given the above constraints.

Sample Input 1

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

Sample Output 1

```
1
```

Sample Input 2

```
6
0 0 0 1 0 0
0 0 0 0 1 1
0 0 0 0 1 1
1 0 0 0 0 0
0 1 1 0 0 0
0 1 1 0 0 0
```

Sample Output 2

```
2
```