Hack It

Time Limit: 2000/1000 MS (Java/Others) Memory Limit: 32768/32768 K (Java/Others)
Total Submission(s): 0 Accepted Submission(s): 0
Special Judge

Problem Description

Tonyfang is a clever student. The teacher is teaching he and other students "bao'sou".
The teacher drew an n*n matrix with zero or one filled in every grid, he wanted to judge if there is a rectangle with 1 filled in each of 4 corners.
He wrote the following pseudocode and claim it runs in $O(n^2)$:

```
let count be a 2d array filled with 0s
iterate through all 1s in the matrix:
  suppose this 1 lies in grid(x,y)
  iterate every row r:
    if grid(r,y)=1:
      ++count[min(r,x)][max(r,x)]
      if count[min(r,x)][max(r,x)]>1:
        claim there is a rectangle satisfying the condition
claim there isn't any rectangle satisfying the condition
```

As a clever student, Tonyfang found the complexity is obviously wrong. But he is too lazy to generate datas, so now it's your turn.
Please hack the above code with an n*n matrix filled with zero or one without any rectangle with 1 filled in all 4 corners.
Your constructed matrix should satisfy $1 \leq n \leq 2000$ and number of 1s not less than 85000.

Input

Nothing.

Output

The first line should be one positive integer n where $1 \leq n \leq 2000$.

n lines following, each line contains only a string of length n consisted of zero and one.

Sample Input

```
(nothing here)
```

Sample Output

```
3
010
000
000
(obviously it's not a correct output, it's just used for showing output format)
```

Statistic | Submit | Clarifications | Back