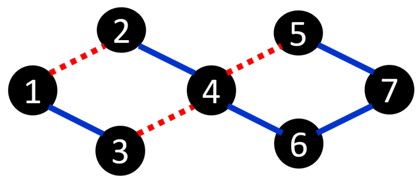
Probem Set 14
15-295 Spring 2020

Problem A
Coloring Contention
Time limit: 1 second



Alice and Bob are playing a game on a simple connected graph with N nodes and M edges.

Alice colors each edge in the graph red or blue.

A path is a sequence of edges where each pair of consecutive edges have a node in common. If the first edge in the pair is of a different color than the second edge, then that is a "color change."

After Alice colors the graph, Bob chooses a path that begins at node 1 and ends at node N. He can choose any path on the graph, but he wants to minimize the number of color changes in the path. Alice wants to choose an edge coloring to maximize the number of color changes Bob must make. What is the maximum number of color changes she can force Bob to make, regardless of which path he chooses?

Input

The first line contains two integer values N and M with $2 \leq N \leq 100\,000$ and $1 \leq M \leq 100\,000$. The next M lines contain two integers a_i and b_i indicating an undirected edge between nodes a_i and b_i ($1 \leq a_i, b_i \leq N$, $a_i \neq b_i$).

All edges in the graph are unique.

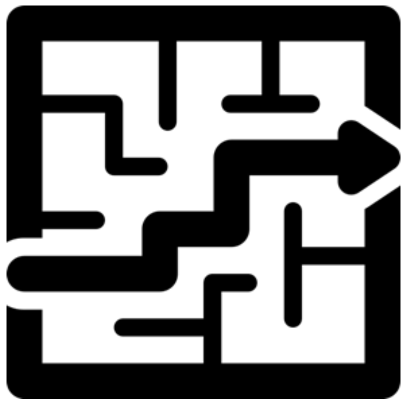
Output

Output the maximum number of color changes Alice can force Bob to make on his route from node 1 to node N.

Sample Input 1	Sample Output 1
3 3 1 3 1 2 2 3	0

Sample Input 2	Sample Output 2
7 8 1 2 1 3 2 4 3 4 4 5 4 6 5 7 6 7	3

Problem B
Maze Connect
Time limit: 5 seconds



Given an orthogonal maze rotated 45 degrees and drawn with forward and backward slash characters (see below), determine the minimum number of walls that need to be removed to ensure it is possible to escape outside of the maze from every square of the (possibly disconnected) maze.

```
/\
\/
```

This maze has only a single square fully enclosed. Removing any wall will connect it to the outside.

```
/\..
\.\.
.\/\
..\/
```

This maze has two enclosed areas. Two walls need to be removed to connect all squares to the outside.

Input

The first line has two numbers, R and C, giving the number of rows and columns in the maze's input description. Following this will be R lines each with C characters, consisting only of the characters '`/`', '`\`', and '`.`'. Both R and C are in the range $1 \ldots 1\,000$.

Define an odd (even) square as one where the sum of the x and y coordinates is odd (even). Either all forward slashes will be in the odd squares and all backslashes in the even squares, or vice versa.

Output

Output on a single line an integer indicating how many walls need to be removed so escape is possible from every square in the maze.

Examples

Sample Input 1

```
2 2
/\
\/
```

Sample Output 1

```
1
```

Sample Input 2

```
4 4
/\..
\.\.
.\/\
..\/
```

Sample Output 2

```
2
```

Sample Input 3

```
2 2
\/
/\
```

Sample Output 3

```
0
```

Sample Input 4

```
8 20
/\/\/\/\/\/\/\/\/\/\
\../\.\/./././\/\/\/
/./\.././\/\.\/\/\/\
\/\/\.\/\/./\/..\../
/\/./\/\/./..\/\/..\
\.\.././\.\/\/./\.\/
/.../\../..\/./.../\
\/\/\/\/\/\/\/\/\/\/
```

Sample Output 4

```
26
```

C Finite Fractions

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Given n decimal fractions. Find the least integer B such that the given n fractions all have finite number of digits in base-B.

Input

The first line contains a single integer n ($1 \le n \le 1000$).

Each of the following lines contains two integers a and b ($1 \le a, b \le 1000000000$), which represents a decimal fraction $\frac{a}{b}$.

Output

The smallest integer B such that the given n fractions all have finite number of digits in base-B.

Since B could be large, display B in decimal modulo $10^9 + 7$.

Example

standard input	standard output
3 3 99 1 99 1 11	33

Note

A base-B fraction $abc.def = a \times B^3 + b \times B^2 + c \times B^1 + d \times B^{-1} + e \times B^{-2} + f \times B^{-3}$.

Problem D
Error Correction
Time limit: 1 second

You are given W, a set of N words that are anagrams of each other. There are no duplicate letters in any word. A set of words $S \subseteq W$ is called "swap-free" if there is no way to turn a word $x \in S$ into another word $y \in S$ by swapping only a single pair of (not necessarily adjacent) letters in x. Find the size of the largest swap-free set S chosen from the given set W.

Input

The first line of input contains an integer N ($1 \leq N \leq 500$). Following that are N lines each with a single word. Every word contains only lowercase English letters and no duplicate letters. All N words are unique, have at least one letter, and every word is an anagram of every other word.

Output

Output the size of the largest swap-free set.

Sample Input 1	Sample Output 1
6 abc acb cab cba bac bca	3

Sample Input 2

```
11
alerts
alters
artels
estral
laster
ratels
salter
slater
staler
stelar
talers
```

Sample Output 2

```
8
```

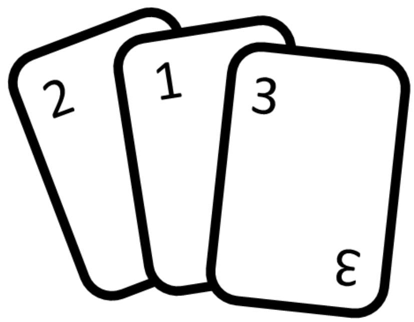
Sample Input 3

```
6
ates
east
eats
etas
sate
teas
```

Sample Output 3

```
4
```

Problem E
Perfect Flush
Time limit: 2 seconds



You are given a list of integers $x_1, x_2, \ldots, x_n$ and a number k. It is guaranteed that each i from 1 to k appears in the list at least once.

Find the lexicographically smallest subsequence of x that contains each integer from 1 to k exactly once.

Input

The first line will contain two integers n and k, with $1 \leq k \leq n \leq 200\,000$. The following n lines will each contain an integer x_i with $1 \leq x_i \leq k$.

Output

Write out on one line, separated by spaces, the lexicographically smallest subsequence of x that has each integer from 1 to k exactly once.

Examples

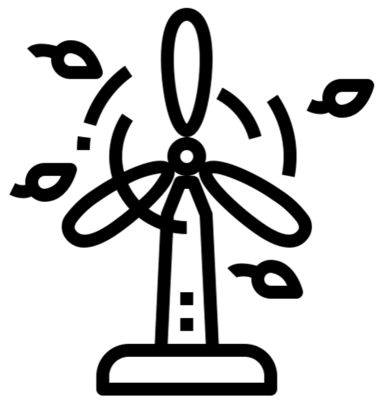
Sample Input 1	Sample Output 1
6 3 3 2 1 3 1 3	2 1 3

Sample Input 2	Sample Output 2
10 5 5 4 3 2 1 4 1 1 5 5	3 2 1 4 5

Problem F

Pivoting Points

Time limit: 10 seconds



Consider a set of points P in the plane such that no 3 points are collinear. We construct a "windmill" as follows:

Choose a point p in P and a starting direction such that the line through p in that direction does not intersect any other points in P. Draw that line.

Slowly rotate the line clockwise like a windmill about the point p as its pivot until the line intersects another point p' in P. Designate that point p' to be the new pivot (call this "promoting" the point p'), and then continue the rotation.

Continue this process until the line has rotated a full 360 degrees, returning to its original direction (it can be shown that the line will also return to its original position after a 360 degree rotation).

During this process, a given point can be promoted multiple times. Considering all possible starting pivots and orientations, find the maximum number of times that a single point can be promoted during a single 360 degree rotation of a line.

Input

The first line of the input will be a single integer n with $2 \leq n \leq 2000$. Following this will be n lines, each with two integers x_i and y_i with $-10\,000 \leq x_i, y_i \leq 10\,000$.

Output

On one line, write an integer with the largest number of times any particular point can be a pivot when an arbitrary starting line does a full rotation as described above.

Examples

Sample Input 1	Sample Output 1
3 -1 0 1 0 0 2	2

Sample Input 2	Sample Output 2
6 0 0 5 0 0 5 5 5 1 2 4 2	3