

Problem A Numbers Painting

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

Dr. Vasechkin wants to paint all numbers from 1 to N in such a way that if number A is divisible by number B , numbers A and B have different colors.

Help Dr. Vasechkin to find such a painting, where the number of the colors used is minimal.

Input

The input contains integer number N ($1 \leq N \leq 10\,000$).

Output

Write the number of the colors M in the desired painting in the first line of the output. In the second line of the output write the desired painting of numbers from 1 to N . The used colors should be represented by numbers from 1 to M . If there are several solutions, choose any of them.

Example

<code>stdin</code>	<code>stdout</code>
12	4 1 2 2 3 2 3 2 4 3 3 2 4

Problem B. Join the Strings

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

His Royal Highness King of Berland Berl XV was a very wise man and had a very accomplished wife, who was aware of the fact, that prominent and outstanding personalities once having written down their names on the pages of glorious History, remain there forever. His Royal Highness King Berl XV experienced an intrinsic, lost nowadays, deep and sincere sense of respect and trust for his beloved spouse. So he decided to acquire a chronicler of his own. Due to the ambiguous nature of misunderstanding and the crying injustice of history to ambiguity, he decided to leave all his royal responsibilities aside and made up his royal mind to find the chronicler, who will make him famous, depicting all his heroic deeds truthfully and gloriously enough.

The King assembled the greatest minds of his kingdom at the Academic Chroniclers Meeting (ACM), as he named it, and decided to test their might. The task was to build the Smallest Lexicographical Concatenation (SLC) out of the given N strings. SLC of N strings s_1, \dots, s_N is the lexicographically smallest their concatenation $s_{i_1} + \dots + s_{i_N}$, where i_1, \dots, i_N is a permutation of integers from 1 through N . It's a great privilege to be a chronicler, so don't miss your chance and don't screw it up! Make the king choose you!

Input

The first line of the input file contains a single integer N ($1 \leq N \leq 100$) indicating the number of strings. The following N lines contain N strings, one string per line. The length of each string is no more than 100 characters. Each string consists only of lowercase Latin letters. There are no any leading or trailing spaces.

Output

Print the SLC of the given N strings to the output file as a single line.

Example

<code>stdin</code>	<code>stdout</code>
6 it looks like an easy problem	aneasyitlikelooksproblem

Problem C Remote Control

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

A couple of years ago Peter Berlandin bought a brand new Berlony TV set. The TV set supports 100 channels, numbered from 0 to 99. The remote control of the TV set has 13 buttons:

```
1 2 3 ↑
4 5 6 ↓
7 8 9
-- 0
```

When pressing the button “↑” the number of the current channel increases by 1 (if the current channel is the 99th, it switches to 0). If you press the button “↓” the number of the current channel decreases by 1 (if the number of the current channel is 0, it switches to the 99th). Switching to the channel from 0 to 9 can be done by pressing the button at the remote control with the corresponding number. Switching to the channel from 10 to 99 can be done by pressing the “--” button followed by two buttons in accordance with the number of the channel required. Casual handling and time don’t do good to any of us! Finally the remote control was damaged and went out of order. Now some of the buttons are not functioning. Your task is to switch from X channel to Y using the minimum number of clicks.

Input

The first 4 lines contain information about the productivity of every button at the remote control. 0 stands for the broken button and 1 stands for the one that is functioning. The first line describes “1”, “2”, “3” and “↑” buttons accordingly. The second line describes “4”, “5”, “6” and “↓” buttons accordingly. The third line describes “7”, “8” and “9” buttons accordingly. The fourth line describes “--” and “0” buttons accordingly. The fifth line describes X and Y.

Output

Print the answer to the problem — the minimum number of clicks to switch from X to Y — in the output file. Print -1 if it is impossible to switch from X to Y.

Examples

stdin	stdout
1 1 1 1 1 1 1 1 1 1 1 1 1 23 52	3
0 0 1 1 1 1 1 1 1 1 1 1 1 23 52	4

Problem D Just Matrix

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

This task is not about Berland or Beerland troubles, roads or flights. There will be no boring coins. This task is about simple square matrix.

Numbers from 1 to n^2 were written down in $n \times n$ square matrix A . Each number was written exactly once. After that for each number the pair $top_{i,j}$ and $left_{i,j}$ was written. $top_{i,j}$ is the number of elements in column j bigger than $A_{i,j}$ and positioned to the top from $A_{i,j}$. $left_{i,j}$ is the number of elements in the row i bigger than $A_{i,j}$ and positioned to the left from $A_{i,j}$.

You are given matrices top and $left$. Your task is to find any possible matrix A fulfilling the requirements of the problem.

Input

The first line of the input contains integer number n ($1 \leq n \leq 600$). Further matrices top and $left$ are written, each in the form of n lines of n non-negative integer numbers. The matrices are separated by the empty line. Numbers in both matrices are not bigger than n .

Output

Write to the output matrix A in the format similar to the input data. If there are several solutions, you can choose any of them. If there is no solution, write to the output just one number 0.

Example

stdin	stdout
3	1 2 6
0 0 0	5 3 7
0 0 0	9 8 4
0 0 2	
0 0 0	
0 1 0	
0 1 2	

Problem E. XOR-omania

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

Professor Vasechkin had a notebook with an outstanding set of non-negative integers A_1, A_2, \dots, A_n . Somehow the most remarkable fact that made this set so outstanding appeared to be the impossibility to find such a subset of two or more elements, that XOR of elements in the subset would equal to zero. One day the professor managed to create a new set of integers $B_1, B_2, \dots, B_{n(n-1)/2}$ through applying the XOR operations to all pairs of elements of A set. The set B was not written in any particular order. Unfortunately due to his natural absent-mindedness professor lost the A set and now he is very confused but still obliged to ask you of a considerable favor. Please restore the set in accordance with the remaining B set.

Input

The first line describes M — the amount of numbers in B set ($1 \leq M \leq 100$, $M = N \times (N - 1)/2$ for some number N). The second line describes M numbers — B_1, B_2, \dots, B_M ($0 \leq B_i \leq 2^{31} - 1$).

Output

Print the A set in one line through a blank. All elements of A should be from 0 to $2^{31} - 1$ inclusively. If there are several solutions of the problem, you can choose any of them. It is guaranteed that there exists at least one A set that satisfies the condition.

Example

<code>stdin</code>	<code>stdout</code>
6 30 19 66 13 92 81	94 64 77 28

Problem F Wolves and Sheep

Input file: `stdin`
Output file: `stdout`
Time limit: 1 second
Memory limit: 64 megabytes

The herds of Berland are in danger! Wolves have attacked a pasture of sheep. The shepherd has decided to exterminate wolves in the neighborhood without causing any damage to the sheep. Thus he took a trophy gun, left to him by his grandfather and headed for the ambush. The gun is cast steel and fires with the armour-piercing shells, and the bullets go right through and can hurt a sheep if a wolf is being shot. The wolves and the sheep are represented by segments. The shepherd is in point $(0, 0)$. The flying path of a bullet is a ray coming from point $(0, 0)$. If the path and the segment, characterizing an animal, intersect — the animal dies. Please find the minimum of shots, that is necessary, to kill all the wolves. We rely upon your prudence, for every sheep should remain safe and sound.

Input

The first line describes two integers N and M ($0 \leq N \leq 10^5$; $0 \leq M \leq 10^5$) — is the amount of the wolves and the sheep accordingly. It is followed by $N + M$ lines. Every line contains four integer numbers X_1, Y_1, X_2, Y_2 ($-1000 \leq X_1, X_2 \leq 1000$; $1 \leq Y_1, Y_2 \leq 1000$), describing the segments. The first N lines describe the disposition of the wolves, the following M lines reveal the situation with the sheep. Segments can degenerate to points.

Output

Print the minimum amount of shots required to kill all the wolves. If you find this request quite impossible to fulfill without killing a single sheep, enter “No solution” (no quotation marks).

Examples

stdin	stdout
1 1 5 5 6 7 3 5 8 5	No solution
2 1 1 1 2 3 -5 4 2 2 999 1000 1000 999	1