Problem A. Marios War

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Here is the field, where each cell can have one of three values:

- "0 an empty cell;
- "1 a cell with the brown mushroom;
- "2 a cell with Mario.

Every minute, Mario kills any brown mushroom that is adjacent to his position (4-directionally), Mario does not move to the empty cell.

How many minutes will it take for Mario to kill brown mushrooms in each cell? The number of Marios can be more than one. If this is impossible, return -1.

Input

The first line contains two integers, m - number of rows of a field, n - number of columns of a field ($1 \leq m, n \leq 1000$). The following m lines contain n values (0, 1, or 2).

Output

Print the minimum time (in minutes) that must elapse until no cell will contain a brown mushroom. If any mushroom is unreachable for every Mario in a field print -1.

Examples

standard input	standard output
3 3 2 1 1 1 1 0 0 1 1	4
3 3 2 1 1 0 1 1 1 0 1	-1
1 3 0 2 0	0

Note

The picture illustrates the procedure of the first example.

In the second example, the mushroom at (3, 1) is unreachable, because Mario can move 4-directionally. So, the output is -1.

In the last example, there are no brown mushrooms, there is nothing to kill, so the answer is 0.

Problem B. Sabina's Game

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

Sabina has found her children's game, where two people are participating. This game has one string which consist of lowercase Latin letters. The winner will be the one who answers the question "Will a whole line disappear?"BEFORE OTHERS. Two identical letters that stand nearby will disappear.

Input

The first line contains a string T (1<=T.size()<=100000), a string of game.

Output

Print "YES"if the line will disappear, otherwise print "NO".

Examples

standard input	standard output
aba	NO
aabb	YES

Problem C. Is it in Matrix?

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

You are given a matrix $m \times n$, which has the following properties:

1. Numbers in each row are sorted in ascending order.
2. The first element of each row is greater than or equal to the last element of the previous row ($a[i][0] \geq a[i-1][n-1]$).

Write an efficient algorithm that searches for a target value, k, in a given matrix.

Input

The first row of input contains two integers m - number of rows, and n - number of columns ($0 \leq m, n \leq 1000$). The further m rows indicate values in each row of a matrix (i.e $a[i][j], for i = 0..m-1, j = 0..n-1$); $-10^4 \leq a[i][j] \leq 10^4$. The last row contains one integer denoting the target value that is need to search ($-10^4 \leq k \leq 10^4$).

Output

Print "true"if k is in a matrix, and "false"if it is not.

Examples

standard input	standard output
4 3 -1 2 4 9 12 15 18 21 30 37 56 73 9	true
4 3 -1 2 4 9 12 15 18 21 30 37 56 73 28	false

Problem D. Substring of a Repeated String

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

Given two strings `A` and `B`. Your task is to find the minimum number of times `A` has to be repeated such that `B` is a substring of it. If `B` cannot be found in `A` after it's extension, return -1. If `A` already contains `B`, the number of repetitions is equal to one by default.

Input

Input contains two lines, where the first line denotes the string `A`, and the second line - string `B`. Input strings contain either lowercase or uppercase letters. Lengths of `A` and `B` are between 1 and 10^5.

Output

Print the minimum number of repetitions of `A`, such that `B` is a substring of `A`.

Examples

standard input	standard output
abcd cdabcdab	3
aaa a	1

Note

For the first test case, answer is 3, because by repeating `A` three times ('*abcdabcdabcd*'), `B` is a substring of it. For the second case, we do not extend the string `A` and `B` is a substring of `A`. Number of repetitions of `A` is 1.

Problem E. Minimal cost

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

An undirected graph without multiple edges and loops is given. It already contains some (possibly zero) number of edges. You can add new edges to it for a fee (the fee is different for each edge). It is required to make the graph connected for the smallest fee.

Input

The first line of input contains a single integer N ($1 \leq N \leq 50$) – the number of vertices in the original graph. In the next N rows are written N nonnegative integers(the j-th number in the i-th line corresponds to the cost of adding an edge connecting the vertices i and j, 0 corresponds to an already existing edge, a positive number – nonexistent), the numbers do not exceed 100. The matrix is symmetric.

Output

Print a single number-the minimum possible cost of the addition of this graph to the connected one.

Example

standard input	standard output
3 0 0 28 0 0 0 28 0 0	0

Problem F. John's graph

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

Mom gave John an undirected graph for his birthday. After some time, he came up with a game with a graph. He takes two vertices and checks to see if they are in the same component.

Input

The first line contains two integers n and m ($2 \le n \le 100000$, $0 \le m \le 99999$), where n is the number of vertices and m is the number of edges. Following m lines contain one edge each in form x, y ($1 \le x, y \le n$), where x, y are edge endpoints. The last line contains two integers s and f ($1 \le s, f \le n$), vertices you have to check.

Output

Print "YES if s and f vertices are in the same component, otherwise print "NO".

Examples

standard input	standard output
3 3 1 2 2 3 3 1 1 3	YES
4 2 1 2 2 3 1 4	NO

Problem G. Rock Game

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

We have a collection of rocks, each rock has a positive integer weight.

Each turn, we choose the two **heaviest** rocks and smash them together. Suppose the stones have weights x and y with x <= y. The result of this smash is:

If x==y, both stones are totally destroyed; If x != y, the stone of weight x is totally destroyed, and the stone of weight y has new weight y-x. At the end, there is at most 1 stone left. Output the weight of this stone (or 0 if there are no stones left.)

Input

1 <= stones.length <= 30; 1 <= stones[i] <= 1000

Output

In a single line print the answer

Example

standard input	standard output
6 2 7 4 1 8 1	1

Note

We combine 7 and 8 to get 1 so the array converts to [2,4,1,1,1] then, we combine 2 and 4 to get 2 so the array converts to [2,1,1,1] then, we combine 2 and 1 to get 1 so the array converts to [1,1,1] then, we combine 1 and 1 to get 0 so the array converts to [1] then that's the value of last stone.

NOTE: Solve with **heap**!

Problem H. One-time guests

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

Jojo given an input stream of N characters consisting only of lower case alphabets. The task is to find the first non repeating character, each time a character is inserted to the stream. If no non repeating element is found print -1. Help Jojo to solve this problem.

Input

The first line of input contains an integer T denoting the no of test cases. Then T test cases follow. Each test case contains an integer N denoting the size of the stream. Then in the next line are x characters which are inserted to the stream.
Constraints:
1 <= T <= 200
1 <= N <= 500

Output

For each test case in a new line print the first non repeating elements separated by spaces present in the stream at every instinct when a character is added to the stream, if no such element is present print -1.

Examples

standard input	standard output
2 4 a a b c 3 a a c	a -1 b b a -1 c
1 6 a d b c a a	a a a a d d

Problem I. Win me if you can!

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

Mark is goint to fight for Fight Club. There were N competitors with different powers. There will be P rounds to fight and in each round Mark's power will be changed. With power M, Mark can kill all the competitors whose power is equal or less than his. Round by round, all the competitors who are dead in previous round will reborn. Such that in each round there will be N competitors to fight. As Mark is tired, please,help him to count the number of competitors that he can win in each round and total sum of their powers.

Input

First line contains number of competitors without Mark. Next line contant powers of these competitors. Line 3 contains number of rounds. The last lines show power of Mark on each round.

$1 \leq \mathrm{N} \leq 10^6$ $1 \leq a[i] \leq 10^3$

Output

Show on each line how many competitors he will win and the sum of their powers.

Example

standard input	standard output
7 7 9 1 8 2 6 2 2 4 8	3 5 6 26

Problem J. Hash of digits

Input file:	standard input
Output file:	standard output
Time limit:	1.5 seconds
Memory limit:	256 megabytes

You have N strings s^i consisting of digits and hashes h^i of these strings, where hash of string s^i equal to h^i. But one day You noticed that hashes and strings were mixed. You need to determine which of the numbers are hashes and which are strings consisting of digits. Also map each string to its hash. Hash should be calculated by formula below. All operations must be performed modulo $1e9 + 7$.

$$h = \sum_{i=0}^{|s|-1} (s_i - 47) \times 11^i$$

Input

Given the number N - number of strings($1 \leq N \leq 10^4$). Following $2 * N$ lines consist from strings and hashes. The length of each string does not exceed 100. It is guaranteed that for each string there is a corresponding hash.

Output

In each line print string with corresponding hash like in example below in order their input.

Examples

standard input	standard output
3 334368200 111 100000 266 123456789 93085	Hash of string "111" is 266 Hash of string "123456789" is 334368200 Hash of string "93085" is 100000
3 13903 338308044731467504464331 3 9839 813695185 425675346 3461762860035486	Hash of string "338308044731467504464331 3" is 425675346 Hash of string "9839" is 13903 Hash of string "3461762860035486" is 813695185

Note

In first example, hash of string "111" was calculated as follows:

$$h = \Big(\big((49 - 47) \times 11^0\big) mod (1e9 + 7) + \big((49 - 47) \times 11^1\big) mod (1e9 + 7)\Big) mod (1e9 + 7) + \\ + \big((49 - 47) \times 11^2\big) mod (1e9 + 7)\Big) mod (1e9 + 7) = 2 + 22 + 242 = 266$$