

Problem A. Queries

Input file: **standard input**
Output file: **standard output**
Time limit: 0.25 seconds
Memory limit: 64 megabytes

Mathematicians are interesting (sometimes, I would say, even crazy) people. For example, my friend, a mathematician, thinks that it is very fun to play with a sequence of integer numbers. He writes the sequence in a row. If he wants he increases one number of the sequence, sometimes it is more interesting to decrease it (do you know why?..) And he likes to add the numbers in the interval $[l; r]$. But showing that he is really cool he adds only numbers which are equal some *mod* (modulo m).

Guess what he asked me, when he knew that I am a programmer? Yep, indeed, he asked me to write a program which could process these queries (n is the length of the sequence):

- **+ p r**
It increases the number with index p by r . ($1 \leq p \leq n$, $1 \leq r \leq 1000000000$)
You have to output the number after the increase.
- **- p r**
It decreases the number with index p by r . ($1 \leq p \leq n$, $1 \leq r \leq 1000000000$) You must not decrease the number if it would become negative.
You have to output the number after the decrease.
- **s l r mod**
You have to output the sum of numbers in the interval $[l; r]$ which are equal *mod* (modulo m). ($1 \leq l \leq r \leq n$) ($0 \leq \text{mod} < m$)

Input

The first line of each test case contains the number of elements of the sequence n and the number m . ($1 \leq n \leq 10000$) ($1 \leq m \leq 10$)

The second line contains n initial numbers of the sequence. ($0 \leq \text{number} \leq 1000000000$)

The third line of each test case contains the number of queries q ($1 \leq q \leq 10000$).

The following q lines contains the queries (one query per line).

Output

Output q lines - the answers to the queries.

Examples

standard input	standard output
3 4	2
1 2 3	3
3	1
s 1 3 2	
+ 2 1	
- 1 2	

Problem B. Personal programming language

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 64 megabytes

Everybody knows that you are not a real programmer until you create your own programming language. Informikas is aware of that too. Finally he is ready to become a real programmer.

Informikas doesn't want to create general purpose language (there are already so many of them!). He is creating a programming language to build text using functions.

The function in the programming language is declared and defined like this:

```
def f = word
```

Here "f" is a function name and "word" is the word returned by the function. For example, the following function would return "hello":

```
def h = hello
```

In addition, functions can be extended. To extend the function you add to function declaration with `<function-name>`. Only functions that are already declared can be used for extension. In the following example function f3 extends functions f1 and f2; function f5 extends f3 and f4; function f6 extends f3 and f2:

```
def f1 = world
def f2 = hello
def f3 with f2 with f1 = say
def f4 = please
def f5 with f3 with f4 = dont
def f6 with f3 with f2 = rather
```

Extending means that the function returns not a single word it defines, but also the words of extended functions. In the previous example calling the function "f3" would return the word defined in "f3" followed by the word defined in "f2" (the first function extended), following by the word defined in "f1" (the second function extended). The result of "f3" would be words "say hello world".

If the function extends other function which is also extending something, then the extensions get propagated all the way down. In the previous example calling "f5" would result in string "dont say hello world please" because after flattening "f3" you would get:

```
f5 with f3 with f2 with f1 with f4
```

However, there is a special case. If the same function gets extended multiple times, only the last occurrence would be left. For example, "f6" would be flattened to:

```
//flattened, but wrong - multiple occurrence of "f2"
```

```
f6 with f3 with f2 with f1 with f2
```

```
//right, leaving only the last occurrence of "f2"
```

```
f6 with f3 with f1 with f2
```

resulting in text "rather say world hello".

Knowing the syntax of the language, could you write a compiler which would compute the text returned?

Input

In the first line of input there is integer N ($1 \leq N \leq 10^5$) – the number of functions defined. In the following N lines there are N functions defined, in the format:

```
def Fi with Gi1 with Gi2 with ...with Gij = Wi
```

where F_i is the name of the function, $G_{i1}, \dots, G_{ij}$ are the names of extended functions, W_i is the word returned by the function F_i . Each of F_i , G_{ij} and W_i consists of $[1;10]$ alphanumeric characters. In the last line there is a string H – the function being called, one of F_i .

It is guaranteed that function is used for extending (positions G_{ij}) only if it was defined (position F_i) above. It is also guaranteed that the word "with" would be used no more than 10^5 times. In addition, words "def" and "with" are reserved and won't be used in place of neither F_i , G_{ij} nor W_i (however, words "WITH" "wiTH" "Def etc. are not reserved).

Output

Output the string which would be printed after calling function H .

Examples

standard input	standard output
4 def a = one def b with a = two def c with a with b = three def d with c = four c	three two one
3 def a = sense def b with a = make def c with b with a with b = doesnt c	doesnt make sense

Note

This problem was inspired by how Scala programming language deals with multiple inheritance.

Problem C. Reordering Ones

Input file: **standard input**
Output file: **standard output**
Time limit: 0.5 seconds
Memory limit: 64 megabytes

Given a board of size $N * M$. The rows are numbered from 1 to N , and the columns are numbered from 1 to M . Each cell of the board contains either 0 or 1.

You can apply 2 types of operations:

1. `swapCol(i, j)`, meaning you can swap 2 different columns
2. `swapRow(i, j)`, meaning you can swap 2 different rows.

You have a function `valid(x1, y1, x2, y2)`, which returns true if and only if every cell in the sub-board with 2 opposite corners (x_1, y_1) , (x_2, y_2) contains all 1.

You need to find maximum value of $(x_0 + y_0)$, such that it is possible to apply the operations, so that `valid(1, 1, x0, y0) = true`. However, everybody is only interested in solutions where $x_0 + y_0 > \max(m, n)$.

Input

On the first line of input there are two integers N and M ($1 \leq N, M \leq 1000$) – the size of the matrix. On the following N lines there are M integers each – the elements of the matrix, either 0 or 1.

Output

On the first line of output print the values x_0 and y_0 giving the maximal possible sum $(x_0 + y_0)$.

On the next line of output print the number R – the number of operations.

On the following R lines output the operations itself in such format `T i j`, where T is the what is swapped (`'C'` for columns and `'R'` for rows), i and j are the indexes of rows/columns being swapped.

If there are multiple solutions, output any of them.

If there is no solution such that $x_0 + y_0 > \max(m, n)$, print single line with "0 0".

Examples

standard input	standard output
3 3 0 1 1 1 1 0 1 1 1	1 3 1 R 1 3
3 1 0 1 1	0 0

Problem D. Xor Sum

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 4 megabytes

You're given an array a with n elements.

Compute the xor sum of the elements that appear an odd number of times in the array a .

Input

The first line contains number n ($1 \leq n \leq 3000000$). The second line of the input contains n numbers, representing the array a . All elements of a are between 0 and 10^9

Output

The only line of the output contains the answer of the problem.

Examples

standard input	standard output
5 4 4 3 3 4	4

Problem E. Slicing cheese

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 64 megabytes

You have a polygon defined by N points (x_i, y_i) and a polyline defined by M points (a_j, b_j) . Polyline cuts polygon into some number of smaller polygons. Find the total perimeter P of resulting polygons. Polyline is guaranteed to start and end outside of the polygon. Each segment of polyline has no more than 1 common point with any segment of polygon segments. All points in the input are distinct. (a_j, b_j) does not lie on any line segment of the polygon.

Input

First line contains integers N and M . ($3 \leq N \leq 1000$, $2 \leq M \leq 1000$)

Next N lines contain integers x_i and y_i .

Next M lines contain integers a_j and b_j .

$(-10^5 \leq x_i, y_i, a_j, b_j \leq 10^5)$

Output

Output a single number P .

The answer is considered correct if its relative or absolute error doesn't exceed 10^{-6} .

Examples

standard input	standard output
4 2 0 0 10 0 10 10 0 10 5 -1 5 11	60.000000000000000

Problem F. Matrix

Input file: **standard input**
Output file: **standard output**
Time limit: 0.5 seconds
Memory limit: 64 megabytes

You're given a matrix with n rows and n columns. A basic property of a square matrix is the main diagonal: all cells that have the same row and column number.

We consider all diagonals that are parallel to the main one. We consider them from left-low corner to the right-upper one. So, the first cell of each diagonal will be, in order: $(n, 1)$ $(n - 1, 1)$... $(1, 1)$ $(1, 2)$... $(1, n)$.

You need to choose one number from each diagonal. More, all $2n-1$ numbers must be pairwise distinct.

Input

The first line contains number n ($1 \leq n \leq 300$). Next n lines contain n numbers, representing the elements of the matrix. All elements of the matrix are between 1 and 10^9 .

Output

If there is no solution, output "NO". Otherwise, output "YES". Next, on the same line, output $2n-1$ numbers, separated by spaces. Each number represents the chosen value from a diagonal. Diagonals are considered in the order given by the problem.

Examples

standard input	standard output
2 1 1 1 1	NO

Problem G. Yet Another Median Task

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 64 megabytes

You're given a matrix a with n lines and n columns. We define the median of an array the element from the middle position of sorted array. If there are two such positions (n is even), get the element whose position is minimal.

Answer q queries of form: what's the median element of a submatrix $(x1, y1, x2, y2)$. Formally, we consider all elements $a[i][j]$ such as $x1 \leq i \leq x2$ and $y1 \leq j \leq y2$ and add them into a temporary array. Then, get the median of the array.

Input

First line of the input contains numbers n ($1 \leq n \leq 800$) and q ($1 \leq q \leq 1000$). Next n lines contain n numbers, describing the 1-based matrix a . All elements from a can fit into an int type. Next q lines contain numbers $x1, y1, x2, y2$, describing a query ($1 \leq x1 \leq x2 \leq n, 1 \leq y1 \leq y2 \leq n$)

Output

Output q lines, each line to answer a query.

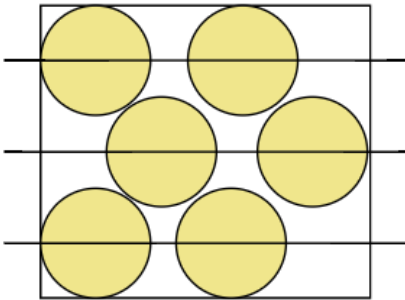
Examples

standard input	standard output
2 4	1
1 2	1
2 2	2
1 1 2 1	2
1 1 1 2	
1 1 2 2	
2 2 2 2	

Problem H. Packing circles

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 64 megabytes

You are given rectangle dimensions (W - width, H - height) and circle diameter D . How many circles can you fit inside that rectangle? Circles must be placed in rows parallel to the bottom. Each row must have the same amount of circles. Each pair of consecutive rows must have the same distance between them. A row is defined by a horizontal line. Circle belongs to a row if its center point lies on that line.



Input

A single line contains 3 integers W , H and D . ($1 \leq W, H, D \leq 10^6$)

Output

A single integer - the maximal amount of circles that can fit inside the rectangle with given restrictions.

Examples

standard input	standard output
2 2 2	1
2 2 1	4

Problem I. Card Jousting

Input file: **standard input**
Output file: **standard output**
Time limit: 0.25 seconds
Memory limit: 32 megabytes

Iahub and Iahubina are playing a card game. Each of them has N cards on which are written numbers. Even if Iahub loves Iahubina very much, he wants to win this game. In order to do that he needs to win all the rounds. A round consists in the choice of the card by Iahubina which will be removed afterwards. Iahub also has to choose a card from his deck, which will be removed afterwards as well. Let B be the card chosen by Iahubina and A the card chosen by Iahub. Iahub can win only if $A|B \geq K$. In addition, Iahub has the possibility to interchange the bits from any two cards, but with a cost. Iahub can interchange the bit i from any card from his deck with the bit j , the cost being $cost[i][j]$. Find the minimum cost which could lead Iahub to win.

Input

The first line of the input contains an integer N ($1 \leq N \leq 200$). The second line contains a vector $A[]$ with N elements, representing the numbers on Iahubina's cards. The third line contains a vector $B[]$ with N elements, representing the numbers on Iahub's cards ($1 \leq A[i], B[i] \leq 256$). On the next line is the integer K ($1 \leq K \leq 127$), followed by 8 lines with 8 values per line, corresponding to the matrix $cost[][]$ ($1 \leq cost[i][j] \leq 1000$, $cost[i][j] = cost[j][i]$).

Output

Output the minimum cost which would lead Iahub to win.

Examples

standard input	standard output
5 127 115 18 81 12 9 73 60 70 19 127 9 7 10 7 5 1 6 1 7 1 9 1 10 2 10 7 10 9 10 9 10 5 10 9 7 1 9 1 5 10 6 7 5 10 10 5 2 3 4 10 1 2 5 10 3 8 5 5 6 10 10 6 4 5 7 1 1 7 9 7 10 5 1 10	10

Problem J. Empty Circle

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

You are given a set of n points $S = \{p_1, p_2, \dots, p_n\}$ on the plane. A triple (p_i, p_j, p_k) is called good if no point in S is strictly inside the circumference of these three points. If it is possible, find a good triple!

Input

The first line contains an integer n , the number of the points ($3 \leq n \leq 10^6$). The i -th of the next n lines contains two integer numbers x_i and y_i , the coordinates of p_i ($-10^4 \leq x_i, y_i \leq 10^4$). No two given points will be equal.

Output

If a good triple exists, output “Yes” in the first line. In the next line output three space-separated integers, the numbers of the three points as they appeared in the input.

If no good triple exists, output “No” in a single line.

Examples

standard input	standard output
4 0 1 0 0 -1 -1 1 -1	Yes 3 1 2
3 -1 -1 0 0 1 1	No
4 -2 -2 2 -2 -2 2 2 2	Yes 1 3 2

Problem K. n -Way Tie

Input file: `stdin`
Output file: `stdout`
Time limit: 0.3 seconds
Memory limit: 256 megabytes

In the country of Tieland, there is an annual tie tying tournament that hosts n participants numbered with integers from 1 to n . Each two participants meet exactly once and have a match in tie tying. There is a winner and a loser in each of the matches. A participant obtains a point for each match victory, and the loser gets zero points for that match. The score of the participant is the total number of points he earned.

However, the jury is worried of the scenario in which all participants have exactly the same score; in other words, there is an n -way tie. In that case any participant would be both the winner and the loser of the competition, which is absurd! Alas, the jury is tied up with the organisation of the tournament; help them and find out whether such a situation might occur!

Input

The single line of the input contains an integer n ($1 \leq n \leq 1000$), the number of participants.

Output

If such a scenario is possible, output “Yes” in the first line. Then output $n - 1$ lines, the i -th containing $n - i$ space-separated integers. The j -th number of the i -th line a_{ij} should denote the outcome of the match between participants with numbers i and $i + j$:

- $a_{ij} = 1$, if the i -th participant won the match;
- $a_{ij} = 0$, if the $(i + j)$ -th participant won the match.

If multiple solutions exists, output any of them.

If such a scenario is not possible, output “No” in a single line.

Examples

stdin	stdout
2	No
3	Yes 1 0 1

Problem L. Basketball

Input file: **standard input**
Output file: **standard output**
Time limit: 0.25 seconds
Memory limit: 64 megabytes

The most important part of playing basketball in the yard is to choose two equally strong teams. Otherwise, the game could be ruined because the stronger team could win easily.

However, sometimes the number of players in the team is not so important. Especially, if tall guys are playing with small kids.

There are two teams: Reds and Blues. For each player in the teams we know his skills.

We are interested in how many ways we can choose some players from Reds and some players from Blues, so that the created teams are equally strong. The strength of the team is the sum of the skills of its individual players.

Input

The first line of each test case contains the number of players in Reds n and the number of players in Blues m ($1 \leq n, m$) ($n + m \leq 17$).

The second line of each test case contains the skills of players in Reds from 1 through n .

The third line of each test case contains the skills of players in Blues from 1 through m .

Skills are integers in the interval $[1; 100000000]$.

Output

Output the number of ways of choosing two equally strong teams.

Examples

standard input	standard output
1 2 2 1 1	1
1 1 1 2	0