Problem A. Answers

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

Given two sequences $\{T(1), T(2), \ldots, T(N)\}$ and $\{C(1), C(2), \ldots, C(N)\}$. Your task is to answer Q questions of the following form: if there exists a sequence $\{x_1, x_2, \ldots, x_k\}$ such as

- $\forall 2 \le i \le k, T(x_{i-1}) = x_i$,
- $\sum_{1 \le i \le k} C(x_i) = M_j$?

Input

The first line of the input contains two integers N and Q. The second line contains N integers $T(1)$, $T(2)$, …, $T(N)$. The third line contains N integers $C(1), C(2), \ldots, C(N)$. The fourth line contains Q integers $M_1, M_2, \ldots, M_Q$.
$(1 \le N, Q \le 10^5, 1 \le T(1), T(2), \ldots, T(N) \le N, 1 \le C(1), C(2), \ldots, C(N) \le 2, |M_1|, |M_2|, \ldots, |M_Q| \le 10^9)$

Output

For each question, print "YES" or "NO".

Example

standard input	standard output
2 4 2 2 2 1 0 1 2 3	NO YES YES YES

Problem B. Business

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	256 mebibytes

Businessman Dogy lives in city S, which consists of n districts. There are $n-1$ bidirectional roads in city S, each connects a pair of districts. Indeed, city S is connected, i.e. people can travel between every pair of districts by roads.

In some districts there are shops founded by Dogy's competitors. when people go to shops, they'll choose the nearest one. In cases there are more than one nearest shops, they'll choose the one with minimal city number.

Dogy's money could support him to build only **one** new shop, he wants to attract as many people as possible, that is, to build his shop in some way that maximize the number of people who will choose his shop as favorite. Could you help him?

Input

There are no more than 420 test cases. Please process till EOF.

In each test case:

First line: an integer n indicating the number of districts.

Next n-1 lines: each contains three numbers b_i, e_i and w_i, $(1 \le b_i, e_i \le n, 1 \le w_i \le 10000)$, indicates that there's one road connecting city b_i and e_i, and its length is w_i.

Last line : $n (1 \le n \le 10^5)$ numbers, each number is either 0 or 1, i-th number is 1 indicates that the i-th district has shop in the beginning and vice versa.

You may assume that sum of n in all test cases does not exceed 333 333.

Output

For each test case, output one number, denotes the number of people you can attract, taking district as a unit.

Example

standard input	standard output
5 1 2 1 2 3 1 3 4 1 4 5 1 1 0 0 0 1 5 1 2 1 2 3 1 3 4 1 4 5 1 1 0 0 0 0 1 1 1 0	2 4 0 1

Problem C. Curves and Symmetries

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

A symmetric polynomial is a polynomial in n variables that remains the same polynomial under any permutation of the variables. For example, $f(x_1, \ldots, x_n) = x_1 + \ldots + x_n$ is a symmetric polynomial (it is in fact called the first elementary symmetric polynomial). Symmetric polynomials have many important applications. They can, for instance, be used to prove that there is no formula for the roots of a general five-degree polynomial.

In this problem however, we will concern ourselves with another kind of symmetry. Consider an infinite curve P in the plane where both x and y coordinates are given by polynomials, i.e.,

$$\begin{aligned} x(t) &= a_n t^n + a_{n-1} t^{n-1} + \ldots + a_1 t + a_0, \\ y(t) &= b_m t^m + b_{m-1} t^{m-1} + \ldots + b_1 t + b_0. \end{aligned}$$

We say that such a curve is symmetric around a straight line L (given as an equation $Ax + By + C = 0$) if there exists a real number t_0 such that for all $t \in \mathbb{R}$ the point $(x(t_0 + t), y(t_0 + t))$ is the reflection of $(x(t_0 - t), y(t_0 - t))$ around the line L, and we call the line L a symmetry line for the curve P. For example, consider the curve P given by

$$\begin{aligned} x(t) &= -5t^5 - 26t^4 - 19t^3 + 59t^2 + 111t + 26, \\ y(t) &= -t^5 + 17t^3 - 9t^2 - 61t + 12. \end{aligned}$$

This curve is symmetric around the line $5x + y + 92 = 0$ with $t_0 = -1$ (see Figure 1).

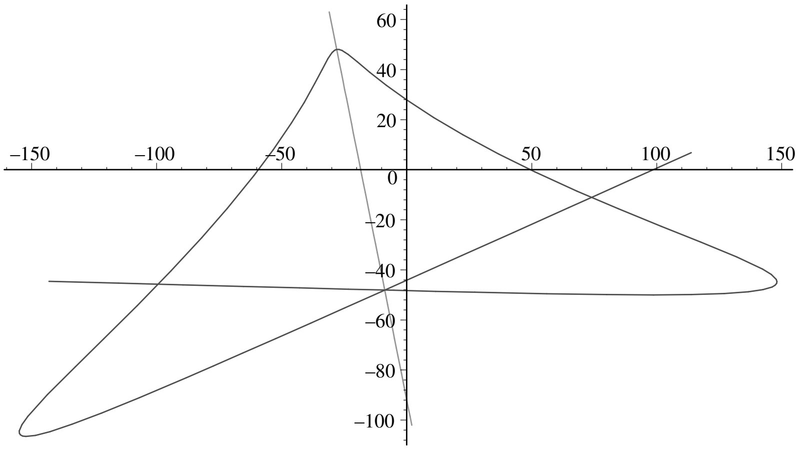

Figure 1: Illustration of Sample Input 1, drawn from $t = -3.9$ to $t = 1.9$.

Now, your task is to write a program that, given the two polynomials $x(t)$ and $y(t)$, finds a symmetry line of the curve (if one exists).

Input

The first line of input contains an integer n ($0 \le n \le 10$), the degree of x. Then follows a line with $n + 1$ integers $a_n, \ldots, a_1, a_0$, where a_i is the degree i coefficient of x. Then follow two lines describing the polynomial y in the same format.

If either of $x(t)$ or $y(t)$ is the zero polynomial, its degree is given as 0. The coefficients have absolute values bounded by 1 000. You may assume that the leading coefficient a_n of each polynomial is non-zero, except in the case when the polynomial is the zero polynomial.

Output

Output three real numbers A, B and C, indicating that $Ax + By + C = 0$ is a symmetry line for the given curve. If there is no symmetry line, let $A = B = C = 0$. If the curve has more than one symmetry line, any one will be accepted.

In the case when a symmetry line exists, the provided line must satisfy the following conditions:

- $0.5 \le \max(|A|, |B|) \le 10^{100}$.
- The direction of the provided line is within 10^{-6} radians of some symmetry line.
- The value of $\frac{C}{\max(|A|,|B|)}$ is correct within an absolute or relative error of 10^{-6}.

Example

standard input	standard output
5 -5 -26 -19 59 111 26 5 -1 0 17 -9 -61 12	5 1 92
1 1 0 3 1 0 0 0	0 0 0
1 1 0 0 0	2.718281828 0 0

Problem D. Distance to Ellipsoid

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

Given a 3-dimension ellipsoid

$$ax^2 + by^2 + cz^2 + dyz + exz + fxy = 1$$

your task is to find the minimal distance between the original point $(0, 0, 0)$ and points on the ellipsoid. The distance between two points (x_1, y_1, z_1) and (x_2, y_2, z_2) is defined as $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$.

Input

There are 100 or less test cases. Please process till EOF.

For each testcase, one line contains 6 real number $a, b, c(0 < a, b, c, < 1), d, e, f(0 \leq e, f, g < 1)$, as described above. **It is guaranteed that the input data forms a ellipsoid**. All numbers are fit in double.

Output

For each test contains one line. A real number describes the minimal distance. Answer will be considered as correct if their absolute error is less than 10^{-5}.

Example

standard input	standard output
`1 0.04 0.01 0 0 0`	`1.0000000`

Problem E. Elections

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

An election selecting the members of the parliament in Byteland was held. The only system adopted in this country is the party-list proportional representation. In this system, each citizen votes for a political party, and the number of seats a party wins will be proportional to the number of votes it receives. Since the total number of seats in the parliament is an integer, of course, it is often impossible to allocate seats exactly proportionaly. In Byteland, the following method, known as the D'Hondt method, is used to determine the number of seats for each party.

Assume that every party has an unlimited supply of candidates and the candidates of each party are ordered in some way. To the y-th candidate of a party which received x votes, assign the value x/y. Then all the candidates are sorted in the decreasing order of their assigned values. The first T candidates are considered to win, where T is the total number of seats, and the number of seats a party win is the number of its winning candidates.

The table below shows an example with three parties. The first party received 40 votes, the second 60 votes, and the third 30 votes. If the total number of seats is $T = 9$, the first party will win 3 seats, the second 4 seats, and the third 2 seats.

# of votes	x/1	x/2	x/3	x/4	x/5	x/6	x/
40	40	20	13.33...	10	8	6.66...	5.7
60	60	30	20	15	12	10	6.57
30	30	15	10	7.5	6	5	3.28

When selecting winning candidates, ties are broken by lottery; any tied candidates will have a chance to win. For instance, in the example above, if $T = 5$ then two candidates tie for the value 20 and there are two possible outcomes:

- The first party wins 2 seats, the second 2 seats, and the third 1 seat.
- The first party wins 1 seat, the second 3 seats, and the third 1 seat.

You have just heard the results of the election on TV. Knowing the total number of valid votes and the number of seats each party won, you wonder how many votes each party received.

Given N, M, and S_i ($1 \le i \le M$), denoting the total number of valid votes, the number of parties, and the number of seats the i-th party won, respectively, your task is to determine for each party the minimum and the maximum possible number of votes it received. Note that for some cases there might be no such situation with the given N, M, and S_i.

Input

The first line of the input contains two integers N ($1 \le N \le 10^9$) and M ($1 \le M \le 3 \cdot 10^4$), where N is the total number of valid votes and M is the number of parties. M lines follow, the i-th of which contains a single integer S_i ($0 \le S_i \le 3 \cdot 10^4$), representing the number of seats the i-th party won. You can assume that there exists i with $S_i \neq 0$.

Output

If there is no situation with the given N, M, and S_i, display the word "`impossible`". Otherwise, output M lines, each containing two integers. The first integer and the second integer in the i-th line should be the minimum and the maximum possible number of votes the i-th party received, respectively.

Examples

standard input	standard output
10 2 2 1	5 7 3 5
5 6 0 2 0 2 6 0	0 0 1 1 0 0 1 1 3 3 0 0
2000 5 200 201 202 203 204	396 397 397 399 399 401 401 403 403 404
15 10 1 1 1 1 1 1 1 1 1 13	impossible
1000000000 9 12507 16653 26746 21516 29090 10215 28375 21379 18494	67611619 67619582 90024490 90033301 144586260 144597136 116313392 116323198 157257695 157269050 55221291 55228786 153392475 153403684 115572783 115582561 99976756 99985943

Problem F. Fruit Game

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

Alice and Bob are playing game. Alice has A yuan, and Bob has B yuan when the game starts.

There are N fruits to be purchased. Fruit i costs C_i yuan. Alice and Bob take turn to buy some (≥ 1) fruits. The player who cannot do this turn lose.

There is some special rule — player can buy fruit i only if fruit $(i-1)$ has been purchased for $i > 1$.

Alice's turn is first. Who will win assuming that both players are playing optimally?

Input

The first line of the input contains three integers N, A, B. The second line contains N integers $C_1, C_2, \ldots, C_N$. $(1 \leq N \leq 10^6, 0 \leq A, B \leq 10^9, 1 \leq C_i \leq 10^9)$

Output

Print "`ALICE`" if Alice wins, or "`BOB`" otherwise.

Examples

standard input	standard output
3 2 2 1 2 1	ALICE
3 2 3 1 2 1	BOB

Problem G. Guards on the Wall

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

You are an IT system administrator in the Ministry of Defense of Polygon Country.

Polygon Country's border forms the polygon with N vertices. Drawn on the 2D-plane, all of its vertices are at the lattice points and all of its edges are parallel with either the x-axis or the y-axis.

In order to prevent enemies from invading the country, it is surrounded by very strong defense walls along its border. However, on the vertices, the junctions of walls have unavoidable structural weaknesses. Therefore, enemies might attack and invade from the vertices.

To observe the vertices and find an invasion by enemies as soon as possible, the ministry decided to hire some guards. The ministry plans to locate them on some vertices such that all the vertices are observed by at least one guard. A guard at the vertex A can observe a vertex B if the entire segment connecting A and B is inside or on the edge of Polygon Country. Of course, guards can observe the vertices they are located on. And a guard can observe simultaneously all the vertices he or she can observe.

To reduce the defense expense, the ministry wants to minimize the number of guards. Your task is to calculate the minimum number of guards required to observe all the vertices of Polygon Country.

Input

The first line contains an even integer N ($4 \le N < 40$). The following N lines describe the vertices of Polygon Country. Each of the lines contains two integers, X_i and Y_i ($1 \le i \le N$, $|X_i| \le 1,000$, $|Y_i| \le 1,000$), separated by one space. The position of the i-th vertex is (X_i, Y_i).

If i is odd, $X_i = X_{i+1}$, $Y_i \ne Y_{i+1}$. Otherwise, $X_i \ne X_{i+1}$, $Y_i = Y_{i+1}$. Here, we regard that $X_{N+1} = X_1$, and $Y_{N+1} = Y_1$. The vertices are given in counterclockwise order under the coordinate system that the x-axis goes right, and the y-axis goes up. The shape of Polygon Country is simple. That is, each edge doesn't share any points with other edges except that its both end points are shared with its neighbor edges.

Output

Print the minimum number of guards in one line.

Examples

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td><pre>8
0 2
0 0
2 0
2 1
3 1
3 3
1 3
1 2</pre></td><td><pre>1</pre></td></tr>
<tr><td><pre>12
0 0
0 -13
3 -13
3 -10
10 -10
10 10
-1 10
-1 13
-4 13
-4 10
-10 10
-10 0</pre></td><td><pre>2</pre></td></tr>
</table>

Problem H. Happiness and Cities

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	256 mebibytes

There are m visitors coming to visit country A, and they plan to visit all n cities in the country one after another. The cities are numbered from 1 to n by the order they are visited. The visitors start their tour at city 1. Each day, for each visitor i, he has p_i probability to go to next city(which means city number increases by 1), and $1 - p_i$ probability to fall in love with current city and stay there till the end of tour. If a visitor reach city n, he will not move any more.

When visitor i reach city j, he get h_{ij} units of happiness. For $j > 1$, suppose city j is visited by $c_j(c_j > 0)$ visitors and city $j - 1$ is visited by $c_{j-1}(c_{j-1} > 0)$ visitors, then each of city j's visitors will get extra $\frac{c_j}{c_{j-1}}h_{ij}$ units of happiness.

Let h_{tot} denote the total happiness of all visitors at the end of tour. Now you need to calculate the expectation of h_{tot}.

Input

There are no more than 1024 test cases. Please process till EOF.

For each case, the first line contains two integers m and n $(1 \le m, n \le 16)$, indicating the number of visitors and the number of cities respectively.

The second line contains m real numbers $p_i(0 \le p_i \le 1)$—the probability for the i-th visitor to move to next city each day. The probabilities are given with at most 6 digits after decimal point.

Then there are m lines follow, each line contains n integers. The j-th integer of ith line denotes h_{ij} $(1 \le h_{ij} \le 100)$.

You may assume that for all except may be 15 test cases $1 \le n, m \le 8$ holds.

Output

For each test case, print a single real number in a line, represents the expectation of h_{tot}. The answer will be considered valid if it differs from the correct one by at most 10^{-5}.

Example

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td>
<pre>
3 1
0.1 0.2 0.3
10
20
30

3 3
0.5 0.5 0.5
1 1 1
1 1 1
1 1 1

4 4
0.1 0.4 0.2 0.3
7 2 18 10
2 6 9 5
4 4 19 17
7 3 13 17
</pre>
</td><td>
<pre>
60.000000
6.8437500
34.23064559
</pre>
</td></tr>
</table>

Problem I. Ice Age

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

Squirrely has lost his lifetime collection of acorns in an unfortunate geyser accident and now he needs your help! Roll the animals around and get Squirrely to the Acorn, but beware! Not all animals are fluffy as they appear to be. Start from your home forest, and continue through the harsh weather of the snow, the scary swamp and the vast wild west! In your journey you will encounter great dangers, but don't worry! You and Squirrely will be a great team.

Get the Nut is a very cute game as mentioned above. Now Squirrely is in a forest, which can be divided into 6 rows and 8 columns, 48 grids in total. Squirrely wants to get the nut in another grid. However, it is much more difficult than you can image because the forest is full of danger. There are two kinds of other animal in the forest: Mouse and Pig. Pigs are very vicious, they will kill and eat any other animal (except Pigs, of course) in the adjacent grids. Here "adjacent" means two grids share a common edge. Once any other animal enter grids adjacent to a Pig, it will stop and be eaten. Once an animal is adjacent to the nut, it will stop and eat the nut. So if Squirrely is eaten by a Pig or the nut is eaten by a Pig or a Mouse, the game is over and you fail. At the beginning of the game, each animal occupies one grid. For each step, you can choose one animal to roll in either up, down, left or right direction, the animal will keep rolling until it reach the border of the forest or hit a tree or an animal. Your task is to calculate the minimal number of steps should be made that Squirrely can get the nut.

Input

There are no more than 30 test cases. Please process till EOF.

Each test case gives the map of the forest, which has 6 rows and 8 columns. Each grid of the map is one of the following six characters:

1. '`.`' - the empty area which every animal can stay or go through
2. '`#`' - the tree which is an obstacle
3. '`S`' - Squirrely
4. '`M`' - a Mouse
5. '`P`' - a Pig
6. '`N`' - the nut which Squirrely want to get

You should notice that there is one blank line before every test case. You may assume that there is one and only one nut in the forest and the number of animals in the forest will not exceed 5, and the input guarantees that there are at most 32 empty grids(including the original grids occupied by animals) in the map.

Output

For each test case, output a positive integer indicating the desired answer. You may assume that there is always exists a solution to get the nut, and the minimal number of steps should be made will not exceed 30.

Example

standard input	standard output

`#......N`
`###.####`
`###M####`
`###.####`
`###.####`
`###S####`
 	
`########`
`###.####`
`##.....#`
`#S.M.P.N`
`########`
`########`
 	
`########`
`S...M.##`
`####M.##`
`####M.##`
`###P..##`
`####N.##`
 	
`P.....M#`
`...##..#`
`P..##..#`
`.###...#`
`..N#...#`
`.###..S#`
	`4`
	`4`
	`5`
	`15`

Note

Case #1:

1. roll the Mouse at grid (3, 4) up to grid (1, 4).
2. roll the Mouse at grid (1, 4) left to grid (1, 2).
3. roll Squirrely at grid (6, 4) up to grid (1, 4).
4. roll Squirrely at grid (1, 4) right to grid(1, 7) and he can get the nut.

Case #2:

1. roll the Pig at grid (4, 6) up to grid (3, 6).
2. roll the Pig at grid (3, 6) left to grid (3, 4), why grid (3, 4)? Because it will stop to eat the Mouse at grid (4, 4).
3. roll the Pig at grid (3, 4) up to grid (2, 4), you must roll the Pig to grid (2, 4), otherwise, Squirrely will certainly be eaten by the Pig!
4. just roll Squirrely at grid (4, 2) right to grid (4, 7) and he can get the nut.

Case #3:

1. roll the Mouse at grid (4, 5) down to grid (5, 5), and it is eaten by the Pig, notice that the Mouse at grid(5, 5) can not eat the nut because the Mouse will be eaten by the Pig before it can eat the nut!
2. roll the Mouse at grid (3, 5) down.

3. roll the Mouse at grid (2, 5) down.
4. roll Squirrely at grid (2, 1) right to grid (2, 6).
5. roll Squirrely at grid (2, 6) down to grid (6, 6) and he can get the nut.

1. When a Mouse or Squirrely is adjacent to a Pig and a nut at the same time, it will be eaten by the Pig first.
2. At the beginning of the game, it's guaranteed that you can't win or fail at once.

Problem J. Jokémon

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

You are playing the game Jokémon, a common element of which is Jokémon battles. In a battle, you and your opponent each start by sending out a Jokémon of your choice, and then take turns attacking the opposing Jokémon.

Each attack has a type (an integer between 1 and n), and the opposing Jokémon also has either one or two types. Depending on these types, the attack will do different amounts of damage.

When an attack of type i hits a Jokémon with single type j, the attack gets a damage multiplier $a(i, j)$, where a is a type matchup table consisting of entries in $\{0, 0.5, 1, 2\}$. If it hits a Jokémon with double types j and k, it gets a damage multiplier of $a(i, j) \cdot a(i, k)$.

Depending on the value v of the damage multiplier, the game will emit different messages:

$v = 0$	It had no effect.	x
$0 < v < 1$	It's not very effective...	-
$v = 1$	*¡no message¿*	=
$v > 1$	It's super effective!	+

You are new to this game and do not know what the table a looks like. Trying to learn its first row, you have gathered some observations about the game's output when attacking various Jokémon with attacks of type 1. Now you are trying to reconstruct the first row in a way that is consistent with this data.

Input

The first line of input contains two integers n and m ($1 \le n \le 10^5$, $1 \le m \le 10^5$), where n is the number of types and m is the number of observations.

Then follow m lines, each containing two integers i, j and a character c ($1 \le i, j \le n$ and c is one of `x`,`-`,`=` or `+`), where c is the observed effect when attacking a Jokémon with types i and j, as indicated in the table above. If $i = j$, the Jokémon has just a single type.

Output

Output a single line with n characters, each either `x`, `-`, `=` or `+`. The ith character should describe the effect of attacking a Jokémon of type i with an attack of type 1.

If there are multiple valid solutions, you may output any one of them. It is guaranteed that at least one solution exists.

Example

standard input	standard output
5 5 1 2 - 2 4 - 4 5 x 2 3 = 3 4 +	=-+=x