Problem A. Turn on the Light

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

This is an interactive problem.

Putata has n lights numbered from 1 to n from left to right. Initially, all the lights are off. One of the lights is his favorite light, and the number of the light is hidden. Budada wants to know the number of Putata's favorite light, and he can make the following query:

- "`?` x": Turn on the light numbered x if the light numbered x is off, and ask Putata the **absolute value** of the number of lights turned on on the left of Putata's favorite light minus the number of lights turned on on the right. Please notice that you can **not** turn off the light.

Budada can only make no more than 40 queries. Please help him find Putata's favorite light.

In this problem, the interactor is **adaptive**, which means that the answer might not be fixed beforehead and the interactor can select it arbitrarily and the answer will be consistent with your interaction with the interactor.

Input

The first line contains an integer n ($1 \le n \le 10^6$), denoting the number of lights.

Interaction Protocol

You can make no more than 40 queries. To make a query, output "`?` x" ($1 \le x \le n$) on a separate line, then you should read the response from standard input.

In response to the query, the interactor will output the absolute value of the number of lights turned on on the left of Putata's favorite light minus the number of lights turned on on the right.

To give your answer, print "`!` x" ($1 \le x \le n$) on a separate line, where x is the number of Putata's favorite light. The output of the answer is **not** counted towards the limit of 40 queries.

After that, your program should terminate.

After printing a query, do not forget to output end of line and flush the output. To do this, use `fflush(stdout)` or `cout.flush()` in C++, `System.out.flush()` in Java, `flush(output)` in Pascal, or `stdout.flush()` in Python.

Example

standard input	standard output
3	
	? 1
1	
	? 2
2	
	? 3
2	
	! 3

Problem B. Equation Discovering

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

Professor Mika is a researcher in computer science who posed the following question: given n pairs of (x, y) where $1 \le n \le 20$, how can we find the governing equation $y = f(x)$ that fits all the pairs? In other words, he seeks to determine an equation involving binary operators ($+$, $-$, $\times$, $\div$), unary operators (sin, cos), symbol x, and parentheses that fits all the given pairs, e.g., $y = x \times x \div \sin(x)$ or $y = x \times (x + x \div x)$.

To generate all the valid equations, we define a context-free grammar as follows:

1. The start symbol is S.
2. $S \to S + S \mid S - S$
3. $S \to S \times S \mid S \div S$
4. $S \to \sin(S) \mid \cos(S)$
5. $S \to (S) \mid x$

However, to prevent overfitting, we limit the complexity of the equations to be less than or equal to 9, where complexity is defined as two times the number of binary operators ($+$, $-$, $\times$, $\div$) plus one times the number of unary operators (sin, cos) in the equation. For example, the equation $x + (x + x \times x)$ has a complexity of 6, while $x \times \sin(x)$ has a complexity of 3. Only equations with a complexity less than or equal to 9 will be considered correct.

Input

The first line contains one integer n ($1 \le n \le 20$), denoting the number of (x, y) pairs to be fit.

The following n lines, each line two real numbers x, y ($|x|, |y| < 10^3$) with exactly six digits after the decimal point, means the (x, y) pairs.

The value of x is guaranteed to be accurate, and we use some valid equation to generate the value of y, then round it to six digits.

Output

The output has only one line of expression f, consists of '`+`', '`-`', '`*`' (for $\times$), '`/`' (for $\div$), '`sin`', '`cos`', '`x`', '`(`' and '`)`'.

Your answer will be considered correct if it satisfies the following conditions:

1. The expression f is generated using the context-free grammar described earlier and is valid according to that grammar.
2. The expression f has a complexity of no more than 9 and does not exceed 1000 characters in length.
3. For each (x, y) pair, the absolute or relative error between $f(x)$ and y is no greater than 10^{-3}. i.e. $\frac{|f(x)-y|}{\max(1,|y|)} \le 10^{-3}$.
4. When calculating the division operation, the absolute value of the divisor must be no less than 0.01.

The evaluation of $f(x)$ follows the standard mathematical conventions, where expressions within parentheses are evaluated first, followed by multiplication and division (with higher priority than addition and subtraction), and all binary operators are left-associative.

The equation we used to generate the data is guaranteed to satisfy all these four requirements in tests. What's more, to avoid possible floating point problems, for requirement four, we promise the solution meets a stricter bound of 0.02.

Note that there may be multiple valid solutions to this problem. Each of them would be accepted.

Examples

standard input	standard output
3 1.000000 1.000000 2.000000 4.000000 3.000000 9.000000	x*x
3 0.618000 1.517072 0.314000 3.132637 1.414000 0.494016	sin(x)/(x*x)

Problem C. Puzzle: Kusabi

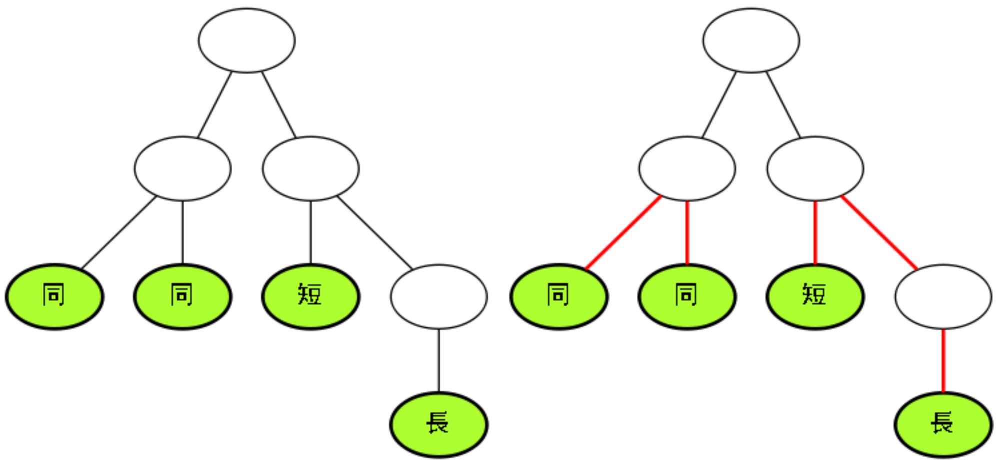
Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Randomly guessing what each symbol corresponds to has only a 16.7% chance of success.

— Freddie Hand

Grammy is a puzzle master. Today, she is playing a variant of "Kusabi" puzzle. In this variant, there is a rooted tree with some Chinese characters on it. The root of the tree is vertex 1, which is not marked. The marked vertices can have a "Chang", "Duan", or "Tong" symbol on it. The goal is to connect all of the marked vertices into pairs such that:

- Each marked vertex is connected to exactly one other marked vertex by marking every edge on the shortest path between them.
- Vertices with character "Chang" must have a longer distance to the root than its counterpart.
- Vertices with character "Duan" must have a shorter distance to the root than its counterpart.
- Vertices with character "Tong" must have the same distance to the root with its counterpart.
- Each **edge** on the tree can be marked at most once.



The left picture illustrates a possible puzzle with only clues, and the right picture shows a possible way to solve the puzzle.

Grammy surely knows how to solve the puzzle, but she decided to give you a quiz. Please solve the puzzle.

Input

The first line contains a single integer n ($1 \le n \le 10^5$), denoting the number of vertices on the tree.

Each of the next $n-1$ lines contains two integers i, p_i ($1 \le p_i < i \le n$) and a string t_i ($t_i \in \{$"`Chang`", "`Duan`", "`Tong`", "`-`"$\}$), denoting that there is an edge between p_i and i, and the type of vertex i is t_i ("`-`" means that vertex i is not marked). It is guaranteed that i is given in increasing order. It is also guaranteed that there is at least one marked vertex.

Output

If the solution does not exist, output "NO" on a single line.

Otherwise, output "YES" on the first line, then output several lines, each of which contains two integers u_i, v_i, denoting a pair of connected vertices in your solution. If there are multiple solutions, output any.

Examples

standard input	standard output
8 2 1 - 3 1 - 4 2 Tong 5 2 Tong 6 3 Duan 7 3 - 8 7 Chang	YES 6 8 5 4
10 2 1 Duan 3 2 Duan 4 2 - 5 4 Chang 6 2 Chang 7 1 Duan 8 6 Tong 9 6 Tong 10 3 Chang	YES 9 8 3 10 2 6 7 5

Problem D. Master of Both III

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

Prof.Chen is the master of theoretical computer science and table tennis. Now he is busy playing table tennis, so he left this algorithm problem for you.

You are given an integer n. For each set S, let $f(S)$ be the minimum sum of costs of operations to change all elements in S into 0. The only operation you can perform on S is:

- Select a subset $T \subseteq S$ and an integer y, and assign $S \leftarrow S \setminus T \cup \{(x + y) \bmod n | x \in T\}$. The cost of this operation is w_y.

For example, assume $S = \{0, 1, 1\}$. If we select $T = \{0, 1\}$ and $y = 2$ to perform the operation, $\{(x+y) \bmod n | x \in T\} = \{2, 0\}$. So S becomes $\{2, 0, 1\}$ after the operation.

Calculate $f(S)$ for each non-empty set $S \subseteq \{0, 1, \ldots, n-1\}$ that does not contains duplicate elements. In order to avoid massive amount of output, output:

$$\sum_{\varnothing \neq S \subseteq \{0,1,\ldots,n-1\}} f(S) \cdot \sum_{v \in S} 2^v$$

Since the answer might be large, output it modulo 998 244 353.

Input

The first line contains one integer n ($1 \leq n \leq 22$), denoting the limit of the set described above.

The second line contains n integers, the i-th integer is w_{i-1} ($1 \leq w_{i-1} \leq 10^9$), denoting the cost of the operations.

Output

Output one integer in one line, the answer.

Examples

standard input	standard output
3 2 1 2	45
4 1919810 999999998 999999997 114114514	152175989

Problem E. Puzzle: Tapa

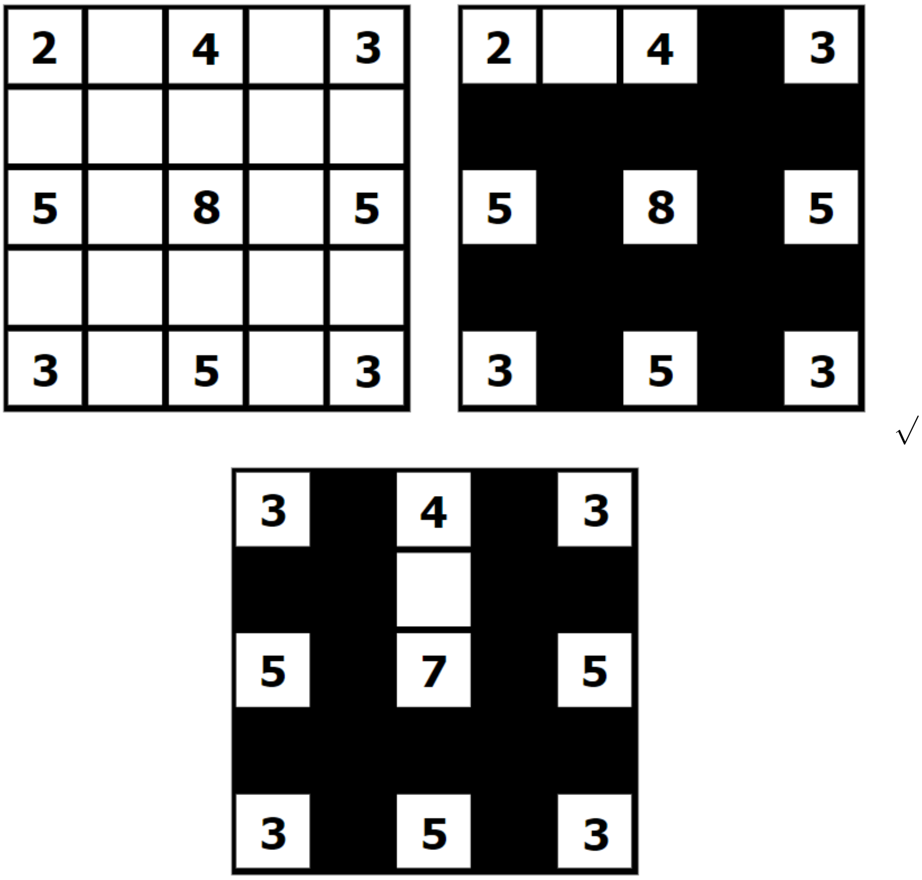
Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 1024 megabytes

A tapa is an appetizer or snack in Spanish cuisine.

— Wikipedia

Grammy is a puzzle master. Today, she is playing a variant of "Tapa" puzzle. In this variant, there are $n \times m$ clues on an $(2n-1) \times (2m-1)$ rectangular grid. All the clues are located on cells (i, j) where i, j are both odd. Each clue is a number that is either equal to or one less than the number of cells around the clue. Specifically, the clues on the corners of the grid can be 2 or 3, the clues on the edges of the grid can be 4 or 5, and the clues on the center of the grid can be 7 or 8. The goal is to shade some cells such that:

- All clue cells are unshaded.
- Each clue cell denotes the number of **consecutive** shaded cells around it.



The top-left picture illustrates a possible 5×5 grid with only clues, the top-right picture shows a possible way to solve the puzzle, and the bottom picture shows a **wrong** solution to a puzzle since the shaded cells around 4 are not consecutive.

Grammy surely knows how to solve the puzzle, but she decided to give you a quiz. Please solve the puzzle.

Input

The first line contains two integers n, m $(2 \le n, m \le 50)$, denoting the size of the grid.

Each of the next $2n-1$ lines contains $2m-1$ characters denoting the grid with given clues. A dot('.') denotes a cell without a clue, while a digit denotes a clue on the cell. It is guaranteed that every cell on the intersection of odd row and odd column has a clue, and all other cells do not contain any clues.

Output

If the solution does not exist, output "NO" on a single line.

Otherwise, output "YES" on the first line, then output $2n-1$ lines, each of which contains $2m-1$ characters, denoting the solution to the puzzle. The format is similar to the input grid, but you should mark the shaded cells with '#'. In other words, a dot('.') in your output denotes an unshaded cell without a clue, a hash('#') denotes a shaded cell, and a digit denotes a clue on the cell.

If there are multiple solutions, output any.

Examples

standard input	standard output
3 3 2.4.3 ..... 5.8.5 ..... 3.5.3	YES 2.4#3 ##### 5#8#5 ##### 3#5#3
3 3 3.4.3 ..... 5.7.5 ..... 3.5.3	NO
2 2 2.2 ... 2.2	YES 2.2 ### 2.2

Problem F. Classic: Classical Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

Deja vu!
I've just been in this place before.

— Dave Rodgers, *Deja Vu*

This is a classical problem, so let's make it quick.

Given a set S of n numbers and a prime number p, find all integer c ($0 \le c < p$) such that $\text{mex}(S_c)$ is maximized where $S_c = \{(c \cdot x) \bmod p | x \in S\}$. Here $\text{mex}(S)$ is the smallest non-negative integer x such that $x \notin S$.

Input

The input contains multiple test cases.

The first line contains an integer T, denoting the number of test cases.

For each test case, the first line contains two integers n, p ($1 \le n \le p \le 2 \times 10^5$), denoting the size of the set and the prime. It is guaranteed that p is a prime.

The following line contains n integers, the i-th integer is a_i ($0 \le a_i < p$), denoting one element in the set. It is guaranteed that $a_i \ne a_j$ if $i \ne j$.

It is guaranteed that the sum of p over all test cases will not exceed 2×10^5.

Output

For each test case, output two integers k, m on the first line, denoting the number of c and the mex.

Output k integers on the following line, denoting the possible c in increasing order.

Example

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

Problem G. Game: Celeste

Input file: `standard input`
Output file: `standard output`
Time limit: 2.5 seconds
Memory limit: 1024 megabytes

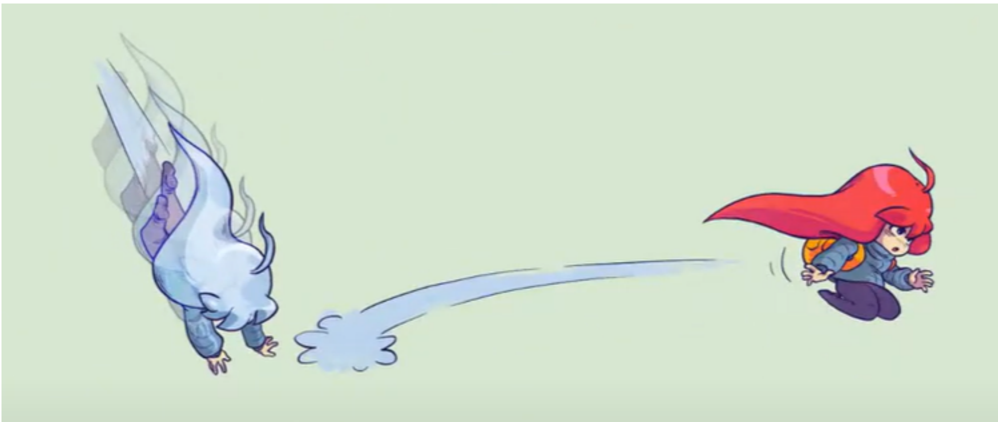
Happy Wavedashing!

— *Wavedash and You*

Madeline is climbing Celeste mountain. The current section contains n pillars located at position $x_1, x_2, \cdots, x_n$. There is no ground between adjacent pillars, so Madeline can only move from pillar to pillar without stopping in the middle. The i-th pillar contains a strawberry of size a_i on top of it. Whenever Madeline is on a pillar, she can choose to collect the strawberry on the pillar.

Madeline has learned the wavedash technique on the first pillar and she wants to practice the technique through this section. However, due to wind conditions, Madeline's wavedash can only bring her to a position in $[x + L, x + R]$, where x is the Madeline's current position.

After arriving at the n-th pillar, Madeline will evaluate her technique using the strawberries she collected. When comparing two sequences of strawberries, she will sort the strawberries by size in non-increasing order and then compare the lexicographic order of the ordered strawberry sequences. She will be more satisfied if the lexicographic order is larger.



Before putting her theory into practice, Madeline wants to plan a route from pillar 1 to pillar n which gives her the maximum satisfaction possible. Please help Madeline.

Input

The input contains multiple test cases.

The first line contains an integer T, denoting the number of test cases.

For each test case, the first line contains three integer n, L, R ($1 \le n \le 10^6, 1 \le L \le R \le 10^9$), denoting the number of pillar and the parameters of wavedash.

The second line contains n integers, the i-th integer is x_i ($1 \le x_i \le 10^9$), denoting the position of the i-th pillar. It is guaranteed that $x_i < x_{i+1}$ for $1 \le i < n$.

The third line contains n integers, the i-th integer is a_i ($1 \le a_i \le n$), denoting the size of the strawberry on the i-th pillar.

It is guaranteed that the sum of n over all test cases is no more than 10^6.

Output

For each test case, if Madeline can't reach the n-th pillar, output -1. Otherwise output two lines.

The first line contains one integer k, the number of strawberries Madeline have collected.

The second line contains k integers, denoting the size of the strawberries Madeline have collected in non-increasing order.

Example

standard input	standard output
2 5 2 3 1 2 3 4 5 5 2 3 1 4 3 1 2 1 4 7 3 3 3	3 5 4 3 -1

Problem H. Classic: N Real DNA Pots

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

First we can ..., then I don't know.

— mysterious oscar

There are n points on the two-dimensional plane. The coordinate of the i-th point is (x_i, y_i). The slope of the segment connecting two points i, j such that $x_i \neq x_j$ is $\frac{y_i - y_j}{x_i - x_j}$.

Please select k points such that the minimum slope of the segment connecting any two points is maximized. Output the minimum slope.

Input

The first line contains two integers n, k ($2 \leq k \leq n \leq 10^5$).

The i-th of the following n lines contains two integer x_i, y_i ($0 \leq x_i, y_i \leq 10^9$). It is guaranteed that $x_i < x_{i+1}$ for $1 \leq i < n$.

Output

Output one real number, denoting the answer.

Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}. Formally, let your answer be a, and the jury's answer be b. Your answer will be considered correct if $\frac{|a-b|}{\max(1,|b|)} \leq 10^{-6}$.

Examples

standard input	standard output
4 3 1 2 2 4 3 3 4 1	-1.0
2 2 1 1 5 3	0.5

Note

Where are the n real DNA pots?

Problem I. MEXimum Spanning Tree

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

Maximum spanning tree is a classical problem in computer science. One day, Grammy came out with a brand new variation of this problem. She wants to find a spanning tree of a weighted graph such that the MEX of the edge weights on the spanning tree is maximized.

The MEX(Minimum EXcluded natrual number) of a set is the minimum natrual number which does not appear in the set. For example, $\text{MEX}(\{0, 2, 4, 5, 7\}) = 1, \text{MEX}(\{0, 1, 2, 3, 6\}) = 4, \text{MEX}(\{3\}) = 0$.

Please help Grammy to solve this problem.

Input

The first line contains two integers n, m ($1 \le n \le 1\,000, 0 \le m \le 1\,000$), denoting the number of vertices and the number of edges.

In each of the following m lines, there are three integers u_i, v_i, w_i ($1 \le u_i, v_i \le n, u_i \ne v_i, 0 \le w_i \le n$), denoting that there is an edge from vertex u_i to vertex v_i with weight w_i.

It is guaranteed that the graph is connected.

Output

Output one integer, denoting the maximum MEX of the spanning tree.

Example

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

Problem J. Master of Polygon

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

Prof.Chen is the master of computational geometry. Now he has a simple polygon with n vertices lying on the Euclidean plane, he would like to give you q queries. In each query, you will be given two points P and Q, you need to check whether the segment PQ intersects with the boundary of the given polygon. Note that even when the segment touches a point of the polygon, you should also answer "`YES`".

Input

The first line of the input contains two integers n and q ($3 \le n \le 200\,000$, $1 \le q \le 200\,000$), denoting the number of vertices and the number of queries.

The next n lines, each line contains two integers x and y ($0 \le x, y \le 30\,000$) that give the coordinates (x, y) of the vertices of the polygon in either clockwise order or counter-clockwise order. The polygon is simple, i.e., its vertices are distinct and no two edges of the polygon intersect or touch, except that consecutive edges touch at their common vertex. In addition, no two consecutive edges are collinear.

Each of the next q lines contains four integers x_1, y_1, x_2 and y_2 ($0 \le x_1, y_1, x_2, y_2 \le 30\,000$), denoting a query segment with endpoint $P(x_1, y_1)$ and $Q(x_2, y_2)$. It's guaranteed that the two endpoints of each segment do not coincide.

Output

For each query, print "`YES`" or "`NO`" in a single line.

Example

standard input	standard output
4 6 1 1 4 1 4 4 1 4 0 2 2 0 0 1 1 1 0 0 5 5 2 2 4 2 2 2 3 2 5 1 0 2	YES YES YES YES NO YES

Problem K. Shuttle Tour

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

Prof.Chen won the champion of the ICPC regional and qualified for the ICPC World Finals next year. To get a good rank in the coming World Finals, he and his teammates are taking part in ICPC Pre-Finals Training Camp in Byteburg now.

Byteburg is a beautiful city with n amazing attraction locations, labeled by $1, 2, \ldots, n$. There are $n-1$ two-way roads between locations such that every pair of locations are connected directly or indirectly. There are at most 50 locations linked to exactly one road in Byteburg.

Unfortunately, some (maybe zero) attractions are not open to tourists. Every afternoon, after 5-hour training, Prof.Chen will browse the tourism website to see which attractions are open, and will then make a shuttle tour plan. You need to perform q operations. Each operation is one of the following:

- "`1` x" ($1 \le x \le n$): Change the status of the attraction at the x-th location. If it is open, it will then be closed, and vice versa.
- "`2` l r" ($1 \le l \le r \le n$): Prof.Chen is now planning to visit every open attraction indexed in $[l, r]$ by shuttle. The shuttle will start at a location, move along roads and finally return to the start location, such that every open attraction is visited at least once. Note that even though the attraction at a location is closed, the shuttle can still reach that location. The start location and the route of the shuttle tour can be assigned by Prof.Chen, he wants to minimize the total length of the route. Please write a program to find the shortest route.

Input

The first line of the input contains two integers n and q ($1 \le n, q \le 200\,000$), denoting the number of locations and the number of operations.

The second line contains a string S of length n, S_i ($S_i \in \{$'`0`', '`1`'$\}$) denoting the initial status of the attraction at the i-th location. Here, '`0`' denotes closed, and '`1`' denotes open.

Each of the next $n-1$ lines contains three integers u_i, v_i and w_i ($1 \le u_i, v_i \le n$, $u_i \ne v_i$, $1 \le w_i \le 10^9$), denoting a two-way road between the u_i-th location and the v_i-th location, whose length is w_i. It is guaranteed that every pair of locations are connected directly or indirectly.

Each of the next q lines describes an operation in formats described in the statement above.

It is guaranteed that there are at most 50 locations linked to exactly one road.

Output

For each query, print a single line containing an integer, denoting the minimum length of the route. Note that when all the attractions indexed in $[l, r]$ are closed, please print "`-1`" instead.

Example

standard input	standard output
5 6 10110 1 2 1 1 3 10 2 4 100 3 5 1000 2 1 5 1 3 2 1 5 2 2 4 2 5 5 2 1 1	222 202 0 -1 0

Problem L. Barkley

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

There are n little pigs in Pigetown. All of them are proficient in competitive programming and the i-th of them has a_i rating. If some of the competitors form a team, the rating of the team will be the greatest common divisor of the rating of the competitors in the team.

Exactly q competitions are going to take place. Pigetown can send exactly one team to participate each contest. For the i-th competition, only the pigs numbered between l_i and r_i have time to participate. Unfortunately, due to a shortage of funds, exactly k_i pigs numbered between l_i and r_i have to work for Putata and Budada to earn funds. Meanwhile, all other pigs in the interval will participate in the contest.

As the coach of Pigetown, for each contest you have to properly select pigs which will participate so that the rating of the team is maximized.

Input

The first line contains two integers n, q ($1 \le n \le 10^5, 1 \le q \le 66666$).

The following line contains n integers, the i-th of them is a_i ($1 \le a_i \le 10^{18}$), denoting the rating of the i-th pig.

The i-th of the following q lines contains three integers l_i, r_i, k_i ($1 \le l_i \le r_i \le n, 1 \le k_i \le \min(3, r_i - l_i)$), denoting one competition.

It is guaranteed that there are no more than 66000 competitions with $k_i = 1$, no more than 660 competitions with $k_i = 2$, and no more than 6 competitions with $k_i = 3$.

Output

Output q lines, the i-th line contains the maximum rating of the teams which participates the i-th contest.

Example

standard input	standard output
4 4 3 2 6 4 1 3 1 2 4 1 1 4 2 1 4 3	3 2 3 6

Problem M. Stage Clear

Input file:	`standard input`
Output file:	`standard output`
Time limit:	7 seconds
Memory limit:	1024 megabytes

Prof.Chen likes playing computer game very much. He is now fighting against scary monsters in the game. The battlefield consists of n intersections, labeled by $1, 2, \ldots, n$. There are m directed arcs between these intersections, such that the battlefield can be regarded as a directed acyclic graph. The player is now at intersection 1 and has X health points (HP).

There is a monster at each intersection except intersection 1. When the player moves to an intersection for the first time, he must fight the monster at that intersection. During the fight, he will lose a_i HP. And when he finally beats the monster, he will be awarded b_i HP. Note that when HP becomes negative (< 0), the game will end, so never let this happen. If the player visits the same intersection more than once, the fight happens only on the first visit, as monsters do not have an extra life. The player can only move along the given m directed arcs, or fly to the intersection 1 by magic. The player can fly for multiple times, and it is always possible for him to reach every intersection.

When all monsters are cleared, Prof.Chen will pass the stage. Please write a program to compute the minimum initial HP to clear the stage.

Input

The first line of the input contains two integers n and m ($n + m \le 72$, $n \ge 2$, $m \ge n - 1$), denoting the number of intersections and the number of directed arcs.

Each of the next $n - 1$ lines contains two integers a_i and b_i ($1 \le a_i, b_i \le 10^{15}$), describing monsters at intersections $2, 3, \ldots, n$.

Each of the next m lines contains two integers u_i and v_i ($1 \le u_i < v_i \le n$), denoting a directed arc from intersection u_i to intersection v_i. There will be at most one arc between a pair of intersections.

It is guaranteed that the player can reach every intersection from the intersection 1.

Output

Print a single line containing an integer, denoting the minimum initial HP required to clear the stage.

Example

standard input	standard output
4 4 4 2 5 3 2 6 1 2 1 3 2 4 3 4	4