Problem A. Agriculture

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

As a member of Japan Agriculture Group, you grow N kinds of plants this year. Each plant has different harvest seasons: the i-th plant must be gathered at some day between s_i and t_i, inclusive.

You plan to gather plants K times, where the j-th gathering day is h_j. On the j-th gethering day, if the i-th plant has not been gathered yet and the gathering day is within the harvest season of the i-th plant, that is $s_i \le h_j \le t_i$, you have to gather the i-th plant.

You are not sure whether your planned days are sufficient to gather all the N plants. If not, you would not be able to survive this cruel Age of Agriculture. Thus you decided to write a program to compute the number of plants gathered after K gathering days you planned.

Input

The first line of the input contains one integer N — the number of plants you will grow ($1 \le N \le 10^5$). The i-th of the following N lines consists of two integers s_i and t_i, which represent that the harvest season of the i-th plant is $[s_i, t_i]$ ($1 \le s_i \le t_i \le 10^9$).

The following line contains the number K of the gathering days you plan ($1 \le K \le 10^5$). The j-th of the following K lines contains an integer h_j ($1 \le h_j \le 10^9$), which is the j-th gathering day you plan. You can assume that holds $h_j < h_{j+1}$ for $1 \le j \le K - 1$.

Output

Print the number of plants gathered after your planned gathering days.

Examples

standard input	standard output
4 1 2 3 3 2 4 7 9 2 3 9	3
4 1 5 5 10 3 8 5 5 1 5	4

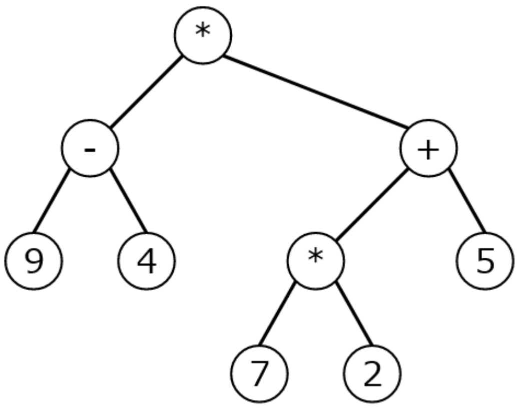
Problem B. Blocks and Expressions

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

To evaluate a program efficiently, a language processor often transforms it into a syntax tree. In this problem you are given a syntax tree of a mathematical expression using ASCII characters. Please evaluate the expression

The syntax tree we consider in this problem is a rooted binary tree where each node has either zero or two children. If a node has zero children, it is an integer node that corresponds to a single integer between 0 and 9, inclusive. On the other hand, if a node has two children, the node is a binary operation node that corresponds to a binary operation of either addition, subtraction or multiplication. In this case the left and right children correspond to the left and right operands of the binary operation, respectively. For example, a figure below represents the syntax tree of expression

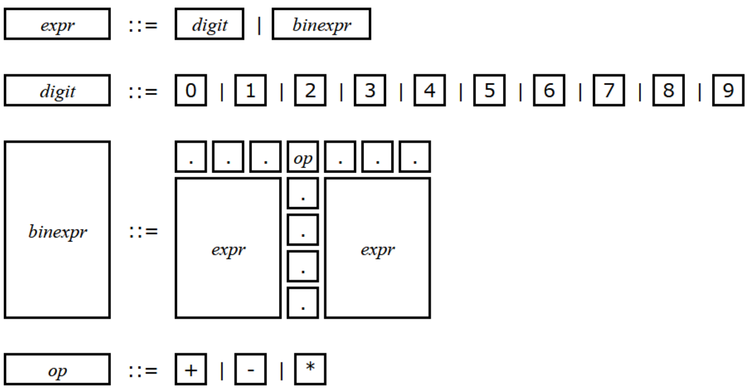
$(9-4)\cdot((7\cdot 2)+5)$.



To represent such a syntax tree using ASCII characters, you are given H strings of W characters. Each character is either '+', '-', '*', a digit between '0' and '9', or a period that represents a blank. For example, here is the representation of the syntax tree of Figure B.1.

```
...*.....
.-.....+.
9.4..*..5
....7.2..
```

Figure below shows the rules (similar to Backus-Naur Form) of such representation of a syntax tree.



More precisely, the rules are defined as follows.

- A *block* is a rectangular region of characters that corresponds to a single node (i.e., either an integer node or a binary operation node) of a syntax tree.
- A block corresponding to an integer node contains only a single digit that is the same integer of the node. The height and width of such a block are 1.
- A block c corresponding to a binary operation node v contains a single operator and two other blocks as children. More precisely, let v_1 and v_2 be the left and right children of the binary operation node, respectively. And let c_1 and c_2 be the blocks that correspond to v_1 and v_2, respectively. The height of c is $\max(h_1, h_2) + 1$ where h_1 and h_2 are the heights of c_1 and c_2, respectively. On the other hand, the width of c is $w_1 + w_2 + 1$ where w_1 and w_2 are the widths of c_1 and c_2, respectively. The topmost row of c consists of w_1 periods followed by an operator followed by w_2 periods where the operator is either '`+`', '`-`' or '`*`'. c_1 is located from the second to the $(h_1 + 1)$-st rows (from the top) and the first to the w_1-st columns (from the left) of c. Similarly, c_2 is located from the second to the $(h_2 + 1)$-st rows (from the top) and the $(w_1 + 2)$-nd to the $(w_1 + w_2 + 1)$-st columns (from the left) of c. Note that although c_1 and c_2 may have different heights, their top borders are always aligned.
- It is guaranteed by the above rules that no two blocks partially overlap each other. In other words, when two blocks overlap, then one of them completely contains the other.
- Any other characters that are not restricted by the above rules are filled by periods.
- The entire region of characters is the "root" block. In other words, the block corresponding to the root node of the syntax tree has height H and width W.

Your task is to calculate the mathematical expression that corresponds to the given syntax tree formatted by the above rules.

Input

The first line of the input contains two integers H and W ($1 \le H, W \le 37$), which represent the height and width of the representation of the given syntax tree. The following H lines consist of strings of length W where each character is either '`+`', '`-`', '`*`', a digit between '`0`' and '`9`', or a period. It is guaranteed that these strings represent a syntax tree of a mathematical expression in a valid form.

Output

Print the calculation result of the mathematical expression that corresponds to the given input.

Examples

standard input	standard output
1 1 5	5
2 3 .-. 9.2	7
4 9 ...*..... .-.....+. 9.4..*..5 ....7.2..	95

Problem C. Changing the Sequences

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

There are sequences $A = (a_1, \ldots, a_N)$ and $B = (b_1, \ldots, b_N)$ with the same length N. a_i denotes the i-th element of A, and its value is an integer between 1 and M, and the same is true for b_j, which is the j-th element of the sequence B.

You can do a magic trick to the sequence A only once: you can prepare a permutation $P = (p_1, \ldots, p_M)$of integers from 1 through M, and can change the sequence A to A' by using P as follows: $a'_i = p_{a_i}$ $(1 \le i \le N)$.

You want to make the distance between the sequence A' and another sequence B closer by changing A to A' through a magic trick.

The *distance* between two sequences is defined as Hamming distance. The Hamming distance between two equal-length sequences is the number of positions at which the corresponding values are different.

Among all possible A', you have to find a sequence which satisfies all of the following conditions.

- No other possible sequences as A' have a smaller distance to B than the distance between this sequence and B.
- It is the lexicographically smallest sequence among possible sequences which has the same distance between B.

Here, a sequence $X = (x_1, x_2, \ldots, x_N)$ is "lexicographically smaller"than another same length sequence $Y = (y_1, y_2, \ldots, y_N)$ if and only if the following condition holds: there exists an index i $(1 \le i \le N)$, such that $x_j = y_j$ for all indices j $(1 \le j < i)$, and $x_i < y_i$.

Input

The first line of the input consists of two integers N $(1 \le N \le 100\,000)$ and M $(1 \le M \le 60)$, which represent that the length of sequences are N, and each sequence has N values between 1 and M.

The second line consists of N integers. The i-th integer is denoted a_i $(1 \le a_i \le M)$.

The third line consists of N integers. The -th integer is denoted b_i $(1 \le b_i \le M)$.

Output

Print N integers, with spaces in between. The i-th integer should be the i-th element of a sequence which satisfies all conditions in the problem statement. Each element of a sequence should be printed as an integer.

Examples

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

Problem D. Determine The Fluctuation Bonus

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

In this unstable world, people who are not afraid of instability will win. Thus it's natural to have a contest that values Most Fluctuated Player (MFP), a player whose rank is most fluctuated during the contest.

International Change Promotion Contest (ICPC) is one of such contests. In this contest, N participants challenge Q quizzes. This contest uses two types of tokens: points and coins. Points are for evaluation of quiz ability itself, but coins are for evaluation of fluctuation. So coins are more important because it directly affects final results.

Each participant initially has 0 points and 0 coins. A participant that answers the i-th quiz gets p_i points. Note that p_i can be negative; it's fine because the focus of the contest is instability (coins), not points. Point ranking is calculated every time after each quiz finishes, and each participant gets coins based on fluctuation of their point rank: if a participant's point rank is changed from a to b the participant gets $|a - b|$ coins, where $|x|$ means the absolute value of x. The point rank of a participant is defined as 1 plus the number of participants who have points (properly) greater than the point of the participant. For example, initially, all the participants have rank 1 since all the participants have 0 points and thus none has points greater than others.

You, as an organizer of ICPC, have a record of a past contest. The record contains information about Q quizzes: the i-th quiz was answered by participant a_i and the point is p_i. But the record doesn't contain the final results: the coins each participant earned in the end. Your task is to write a program to compute the numbers of coins of all the participants after Q quizzes from the record.

Input

The first line contains two numbers N and Q, where N is the number of participants ($1 \le N \le 10^5$) and Q is the number of quizzes ($1 \le Q \le 10^5$). The i-th of the following Q lines consists of two integers a_i and p_i, which represent that the i-th quiz was answered by participant a_i ($1 \le a_i \le N$) and the points of the i-th quiz is p_i ($-10^9 \le p_i \le 10^9$).

Output

Print N lines, the j-th of which is the number of coins the j-th participant earned after all the Q quizzes finish.

Examples

standard input	standard output
3 7 2 -1 1 4 2 5 3 6 1 -7 3 -6 2 9	2 6 5
9 5 2 10 2 -20 2 20 2 -20 2 20	5 32 5 5 5 5 5 5 5
5 10 1 0 3 0 2 0 5 0 4 0 1 0 3 0 2 0 5 0 4 0	0 0 0 0 0

Problem E. Eartheart

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

Eartheart Kingdom is a dimly lit country located underground. Sunlight doesn't usually reach Eartheart Kingdom, but there is one day every few years when sunlight reaches it: SUNDAY. On that day, the sun passes directly above a single hole in the "ceiling"of JAG Kingdom, and the sun peeks out. Bob, the boy in Eartheart Kingdom, is wondering how much time he can see all of the sun through the hole on the next SUNDAY. Please help him.

The ceiling of the Eartheart Kingdom is represented by a two-dimensional plane. The sun is represented by a circle of radius R, and its center moves straight from $(-10^{100}, 0)$ to $(10^{100}, 0)$ by 1 per unit time. The hole is represented by a simple polygon in the two-dimensional plane. Your task is to find the total time that the circle is contained within the polygon. A circle is contained in a polygon if and only if any point inside or on the circumference of the circle is contained inside or on the circumference of the polygon.

Input

The first line contains two integers N and R ($3 \le N \le 10^5$, $1 \le R \le 10^6$) separated by a space, which represent the number of vertices of the polygon and a radius of the circle. Each of the next N lines contains two integers x_i and y_i ($-10^6 \le x_i, y_i \le 10^6$) separated by a space, which represent the i-th vertex of the polygon. The polygon is guaranteed to be simple. In other words, no two edges of the polygon intersect each other.

Output

Print the total time that the circle is contained within the polygon. The answer will be considered as correct if the values output have an absolute or relative error less than 10^{-5}.

Examples

standard input	standard output
4 5 -5 -5 5 -5 5 5 -5 5	0.0000000000
4 5 -10 -10 10 -10 10 10 -10 10	10.000000000
9 10 -100 -80 -90 130 -30 150 0 160 100 130 120 90 110 -60 80 -100 0 -120	190.1569477022

Problem F. Frustration and Bracket Sequences

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	1024 mebibytes

Ryan is interested in strings consisting only of '`(`' and '`)`'. Especially, he loves balanced strings. Any balanced strings can be constructed using the following rules:

- A string "`()`" is balanced.
- The concatenation of two balanced strings is balanced.
- If T is a balanced string, the concatenation of '`(`', T, and '`)`' in this order is balanced.

For example, "`()()`" and "`(()())`" are balanced strings. "`)(`", "`)()(()`" and "`(`" are not balanced strings.

We define Ryan's frustration for a string T as the minimum number of operations required to make T into a balanced string by doing the following operations in any order and any number of times.

- Add '`)`' to the beginning of T.
- Add '`(`' to the end of T.
- Swap two adjacent characters of T.

Ryan has a string S of length N consisting only of '`(`' and '`)`'. Given Q queries, process them in order. There are two kinds of queries with the following formats.

- `1` l r: For each character from the l-th to the r-th (including r-th) of S, if it is '`(`', replace it with '`)`'. If it is '`)`', replace it with '`(`'.
- `2` l r: Output the value of Ryan's frustration for the substring from the l-th through r-th characters of S.

Input

The first line contains two integers N and Q ($2 \leq N \leq 150\,000$, $1 \leq Q \leq 150\,000$) separated by a space, which represent the length of the string S and the number of queries. The following line contains the string S, which consists only of '`(`' and '`)`', and whose length is N. Each of the next Q lines contains three integers t_i, l_i and r_i ($1 \leq t_i \leq 2$, $1 \leq l_i \leq r_i \leq N$) separated by a space, which represent the i-th query. It is guaranteed that there is at least one query with $t_i = 2$.

Output

For each query with $t_i = 2$, print the value of Ryan's sadness, followed by a newline.

Examples

standard input	standard output
6 6 ())()(2 1 6 1 2 4 2 1 4 2 2 5 1 1 5 2 1 6	2 5 0 6
7 5 ((((() 2 1 7 1 1 7 2 1 7 2 3 3 2 2 6	20 26 2 20

Problem G. Good Pizza

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Good Pizza is the famous pizza shop. Since the number of delivery orders has been increasing, it has been busier than before.

One day, several orders came to this shop at the same time! There were N orders, and N customers who placed the order. The i-th customer placed the i-th order.

Through past orders, this shop knows information for each customer. For the i-th customer, it takes t_i hours for delivering the pizza from the shop, and also takes t_i hours for going back to the shop. Moreover, his or her "irritability" is a_i. The bigger irritability a person has, the easier to be angry.

Good Pizza has to deliver orders for customers, and keep their minds on less customer stress. To formulate the amount of stress which the i-th customer feels, let h_i be the time from order to delivery, and let p_i be the number of other customers whose delivery time is faster than that of i-th customer. The amount of stress for the i-th customer (s_i) is formulated as follows:

$s_i = a_i \times (h_i + p_i)$.

For less customer stress, and gaining high satisfaction rates, a better delivery plan is necessary. When you think about it, you must take the following things into account:

- There is only one delivery person.
- A delivery person cannot deliver several orders in parallel. It means that he or she can deliver a single order at once, and when achieved, he or she gets back to the shop for delivering the next order.
- You can assume that it tooks no time for preparing an order, passing it from the shop to a delivery person, and passing it from a delivery person to a customer.

You are the delivery planner of Good Pizza. Above these information and conditions, you have to minimize the total amount of stress for all customers.

Input

The first line of the input contains the number N of customers ($1 \le N \le 100\,000$).

The i-th of the following N lines consists of two integers t_i and a_i, which represent that it takes t_i ($1 \le t_i \le 1\,000$) hours for delivering the pizza from the shop to the i-th customer, and also takes t_i hours for going back to the shop. Moreover, its irritability is a_i ($1 \le a_i \le 1\,000$).

Output

Print the minimum total amount of stress.

Examples

standard input	standard output
3 10 3 3 8 4 2	124
10 17 62 30 79 99 2 88 57 42 46 84 11 44 60 21 98 68 63 17 54	118250

Problem H. Hacks With Includes

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Short code is cool, reasonable, beautiful, and elegant. You love short code. Hence you want to make your code as short as possible. Several techniques are known to make your code shorter. Today, you focus on *includes* in your code.

There are N files you must include to your code. The N files are numbered as 1 through N. Some of them also include others. If file a includes file b and file b includes file c, including a into your code implies including b and c into your code. But including c does not necessarily imply including a or b unless c (indirectly) includes a or b.

You are given information about dependencies of N files, i-th of which describes file a_i includes b_i. From this information, your task is to determine the set of the minimum number of files you have to directly include in order to include all N files indirectly, and output file numbers in the minimum set in ascending order. If such a set is not uniquely determined, output a set with the minimum sum of the file numbers in a set.

Input

The first line of the input consists of two integers N and M, where N is the number of files ($1 \leq N \leq 30\,000$) and M is the number of dependency information ($0 \leq M \leq 5 \times 10^5$). The following M lines represents each dependency, the i-th of which contains two integers a_i and b_i, which means file a_i includes file b_i ($1 \leq a_i \leq N$, $1 \leq b_i \leq N$). There is no duplicate dependency information, i.e. $a_i \neq a_j$ or $b_i \neq b_j$ hold for $1 \leq i < j \leq M$.

Output

Determine the minimum number of files that must be directly included in your code to include all files indirectly, and print file numbers in such a file set in ascending order. If there are multiple sets with the minimum size, output a set with the minimum sum of file numbers.

Examples

standard input	standard output
4 3 2 1 2 4 3 1	2 3
5 6 2 1 2 4 3 1 3 2 5 1 5 2	3 5
9 11 1 3 2 4 2 6 4 1 5 3 5 6 5 8 6 8 7 4 8 1 8 2	5 7 9

Problem I. Impossible-to-finish Race

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

You are a teacher of a class of S students. The students are numbered from 1 to S, and the i-th student has athletic ability A_i and height H_i.

In an upcoming sports day your class is going to compete in a game called impossible-to-finish race. In this race N runners of a team line up in a row, connect their legs using ankle straps (more precisely, connect the first runner's right leg and the second runner's left leg, the second runner's right leg and the third runner's left leg, and so on), and run together toward a goal.

As the teacher of the class you have to choose N students from your class as runners of the race and decide the order of these N runners. Of course, each runner's athletic ability is one of the key factors of the strength of the team. However, you have noticed that if two adjacent runners have very different heights, it ends up losing the strength of the team. After all, if students of numbers $r_1, \ldots, r_N$ line up in this order, the strength of this team is defined as follows.

$$\sum_{i=1}^{N} A_{r_i} - \sum_{i=1}^{N-1} \left| H_{r_i} - H_{r_{i+1}} \right|$$

Your task is to maximize the strength of the team.

Input

The first line of the input contains two integers S and N ($2 \le S \le 10^5$, $2 \le N \le \min(S, 200)$), which represent the number of students in your class and the number of students that you have to choose from your class as runners. Each of the next S lines contains two integers A_i and H_i ($1 \le A_i, H_i \le 10^9$), which represent the athletic ability and height of the i-th student in your class.

Output

Print the maximum strength of the team that you can accomplish.

Examples

standard input	standard output
4 2 40 150 100 185 60 160 80 170	165
4 3 40 150 100 185 60 160 80 170	215
4 3 40 150 100 300 60 160 80 140	160
10 5 31 41 59 26 53 58 97 93 23 84 62 64 33 83 27 95 2 84 19 71	237

Problem J. JAG Graph Isomorphism

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Consider the JAG Graph as the undirected simple connected graph that consists of N vertices numbered from 1 to N and N edges.

Given two JAG graphs G and G'. Are these graphs isomorphic? In other words, is there a permutation $(p_1, \ldots, p_N)$ of $(1, \ldots, N)$ such that G has an edge which connects two vertices u and v if and only if G' has an edge which connects p_u and p_v?

Input

The first line of the input contains a single integer N ($3 \le N \le 2 \times 10^5$), which represents the number of vertices of graphs G and G'. Each of the next N lines contains two integers a_i and b_i ($1 \le a_i, b_i \le N$), which represent that there is an undirected edge connecting vertices a_i and b_i of G. Similarly, each of the next N lines contains two integers c_i and d_i ($1 \le c_i, d_i \le N$), which represent that there is an undirected edge connecting vertices c_i and d_i of G'. You can assume that both G and G' are connected graphs and do not contain self-loops and double edges.

Output

Print "`Yes`" if G and G' are isomorphic. Print "`No`", otherwise.

Examples

standard input	standard output
4 1 2 2 3 2 4 3 4 1 2 1 3 1 4 3 4	Yes
4 1 2 2 3 3 4 1 4 1 2 1 3 1 4 3 4	No
6 1 2 1 3 2 5 2 6 3 5 4 6 1 5 1 6 2 4 2 5 2 6 3 4	Yes

Problem K. King Of Zombies

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Tatsumi, the King of Zombies, planned to form a zombie rock band named Gray Faces in the ICPC-city, and still plans to do so.

But unfortunately, once again, there is only one zombie in ICPC-city. So Tatsumi decided to release the zombie into the city after enhancing the zombie's infectious power, to produce a sufficient number of zombies. The infectious zombie changes a human into a new infectious zombie when the distance between the human and the zombie is less than or equal to D. Note that a zombie that was a human also changes a human into a zombie.

The ICPC-city is represented by an infinitely large two-dimensional plane, and Tatsumi will release the zombie at a point with a coordinate (x_0, y_0). After the release, the zombie will start walking at a speed of $(v_{x,0}, v_{y,0})$ per second. There are also N humans on the two-dimensional plane. When Tatsumi releases the zombie, the i-th human will be at a point with a coordinate $(x_i, y_i)$and will start walking at a speed of $(v_{x,i}, v_{y,i})$ per second. Humans will not change their walking direction or speed when they become zombies.

For each human, Tatsumi wants to know when the human becomes a zombie. Please help him by writing a program that calculates a time when each human becomes a zombie.

Input

The first line of the input contains two integers N and D ($1 \le N \le 10^3$, $0 \le D \le 10^4$) separated by a space, which represent the number of humans and the distance to be infected. The following line contains four integers x_0 y_0 $v_{x,0}$ and $v_{y,0}$ ($-10^4 \le x_0, y_0, v_{x,0}, v_{y,0} \le 10^4$) separated by a space, which represent the initial position and direction of the zombie. Each of the next N lines contains four integers x_i, y_i, $v_{x,i}$ and v_{y_i} ($-10^4 \le x_i, y_i, v_{x,i}, v_{y,i} \le 10^4$) separated by a space, which represent the initial position and direction of the i-th human.

Output

The output consists of N lines. In the i-th line, print the time when the i-th human becomes a zombie. If the i-th human will never become a zombie, print -1 instead. The answer will be considered as correct if the values output have an absolute or relative error less than 10^{-7}.

Examples

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td><pre>5 3
0 0 3 0
10 10 0 -3
1 1 -1 -1
16 1 -1 0
100 100 100 100
-100 -3 10 0</pre></td><td><pre>2.62622655215
0
3
-1
14.2857142857</pre></td></tr>
<tr><td><pre>4 10
0 0 0 0
10 0 0 0
20 0 0 0
30 0 0 0
41 0 0 0</pre></td><td><pre>0
0
0
-1</pre></td></tr>
</table>

Problem L. Lucky Stars Management

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

The company Lucky Stars, where Bill works as the President, has N employees, enumerated from 1 to N in order of their position in the company.

Bill have the employee ID 1. Any other employee, except for the President Bill, have exactly one direct supervisor. The direct supervisor of the employee i is denoted as $P_i < i$. For each of N employees the annual income of this employee is an integer A_j between 0 and $10^9 + 6$.

Once Bill decided to consult with a fortune-teller who is absolutely right about the future of the company and heard something like "someday, K mistakes will be done, and Lucky Stars management became difficult."

Each mistake is caused by exactly 1 employee, and for $i = 1, 2, \ldots, N$ the probability that the mistake is caused by employee i is $1/N$ (note that the mistakes are independent, i.e. it can happen that some employee caused several mistakes).

Bill held a meeting and decided to enforce the personal responsibility in case if fortune-teller was right. Bill wants to find **one** employee and let him pay a fine if K types of mistakes were made by the Lucky Stars personnel at one day. The employee who pays the fine is determined according to the following rules.

- When an employee makes a mistake, not only that employee but also the employee's direct superior, the superior's superior, and so on, are responsible (this rule includes Bill himself).
- Therefore, among the employees who are responsible for all K types of mistakes, the employee with the largest employee number pays the fine.
- Let A_j be the annual salary of the employee who made the j-th mistake for $j = 1, 2, \ldots, K$, and let him be fined $A_1 \times A_2 \times \ldots \times A_K$.

Bill calculated the expected value E_i of the fine paid by the employee number i if the fortune-telling was correct, and represented it modulo $10^9 + 7$.

In other words, for $i = 1, 2, \ldots, N$, the expected fine to be paid by employee number i is expressed as Y_i/X_i (where X_i and Y_i are relatively prime integers and X_i is not divisible by $10^9 + 7$), then answer is Y_i div X_i modulo $10^9 + 7$.

Now, you have the information about structure of the company (i.e. the value of P_i for each employee) and the values of E_i of the expected value of fines, calculated by Bill.

Please determine whether all of the obtained information is likely to be correct, and if so, find the minimum possible Bill's annual salary.

Input

The first line of the input contains two integers N ($1 \le N \le 2 \cdot 10^5$) and K ($1 \le K \le 10^9$, K is odd).

The second line of the input contains $N - 1$ integers $P_2, \ldots, P_N$ ($1 \le P_i \le i - 1$) — the number P_i of the direct supervisor of the employee i. If $N = 1$, the second line is empty.

The third line of the input contains N integers E_i — the calculated expected fines for i-th employee ($0 \le E_i \le 10^9 + 6$).

Output

If the calculation is definitely wrong, print -1. Otherwise print one integer — the minimum possible annual salary of Bill.

Examples

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td><pre>2 1
1
2 1</pre></td><td><pre>4</pre></td></tr>
<tr><td><pre>1 1

2</pre></td><td><pre>2</pre></td></tr>
</table>