Nowcoder Multi-university Contest 2023 #2

Problem List

A	Link with Checksum
B	Link with Railway Company
C	graph
D	The Game of Eating
E	Square
F	Link with Chess Game
G	Link with Centrally Symmetric Strings
H	0 and 1 in BIT
I	Link with Gomoku
J	Smoke
K	Box
L	Link doesn't want to cut tree
M	Fundamental Skills in Data Structures

2023/7/21

Problem A. Link with Checksum

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

Link has invented an algorithm called CRC-LINK. In order to calculate CRC-LINK checksum of a datagram, let's assume the input byte array as D and the CRC-LINK polynomial as P ($P = $ 0x04C11DB7), and follow the three steps below.

1. Treat each byte of the input byte array D as an 8-bit binary number. Append 32 zero bits to the end of D, creating an initial remainder R.
2. Repeat the following step for N times, where N is the number of bits in D.
   - If the most significant bit of R is 1, shift R one bit to the left, and then perform an XOR operation: XOR the most significant 32 bits of R with the CRC-LINK polynomial P.
   - Otherwise, just shift R one bit to the left.
3. The most significant 32 bits of R is the CRC-LINK checksum of D.

For example, the CRC-LINK checksum of byte array {0x01, 0x02} can be calculated by:

```
00000001 00000010 00000000 00000000 00000000 00000000
(shift left for 7 bits)
10000001 00000000 00000000 00000000 00000000 00000000
(shift left for 1 bit, then xor P)
00000110 11000001 00011101 10110111 00000000 00000000
(shift left for 5 bits)
11011000 00100011 10110110 11100000 00000000 00000000
(shift left for 1 bit, then xor P)
10110100 10000110 01110000 01110111 00000000 00000000
(shift left for 1 bit, then xor P)
01101101 11001101 11111101 01011001 00000000 00000000
(shift left for 1 bit)
11011011 10011011 11111010 10110010 00000000 00000000
(since we have shifted 16 bits in total, the left 32 bits above is what we need)
```

Please note that CRC-LINK may be slightly different from CRC32.

For a datagram consisting of three parts: a header (n_1 bytes), a checksum (4 bytes), and a footer (n_2 bytes), the header and footer are given, while the checksum needs to be determined by you. You should choose a checksum that matches the CRC-LINK calculation result of the entire datagram. Formally, you should make sure CRC-LINK(concat(header,YourAnswer,footer)) equals to YourAnswer.

You may refer the example explanation for better understanding of how to check your answer.

Input

The first line contains two integers, n_1 and n_2 ($1 \leq n_1, n_2 \leq 10^5$), representing the lengths of the header and footer, respectively.

The second line contains n_1 integers, where the i-th integer a_i $(0 \le a_i < 2^8)$ represents the value of the i-th byte in the header.

The third line contains n_2 integers, where the i-th integer b_i $(0 \le b_i < 2^8)$ represents the value of the i-th byte in the footer.

Output

If a feasible checksum exists, output the checksum value (in decimal). Otherwise, output '-1'.

If multiple feasible solutions exist, you can output any one of them.

Example

standard input	standard output
1 2 114 51 4	456567105

Note

In the first example, when outputting '456567105', the answer is correct because CRC-LINK(72 1B 36 A9 41 33 04) equals to '456567105'.

Problem B. Link with Railway Company

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 second
Memory limit:	256 megabytes

Link is the owner of a railway company. The railway company currently operates n stations and m railway lines.

The railway network consists of $n-1$ bidirectional railways, and any two stations can be reached from each other through the railway network. The railways in the network can be represented by $n-1$ triplets (u_i, v_i, c_i), where u_i and v_i are the starting and ending stations of the railway, and c_i is the daily maintenance cost for that railway.

The operated transport lines of the company can be represented by m quadruplets (a_i, b_i, x_i, y_i). Here, a_i and b_i are the starting and ending stations of the line, x_i represents the daily revenue generated by the line, and y_i represents the additional expenses incurred by operating this line daily.

The company is currently undergoing a reform: abandoning some underperforming lines and stopping maintenance on certain railways to improve the company's profitability.

Link wants to know, in the optimal state, what is the maximum daily revenue the company can achieve after the reform?

Input

The first line contains two integers, n and m $(2 \le n \le 10^4, 1 \le m \le 10^4)$.

The next $n-1$ lines contain three integers each, u_i, v_i, and c_i $(1 \le u_i, v_i \le n, u_i \ne v_i, 1 \le c_i \le 10^5)$, representing a railway.

The next m lines contain four integers each, a_i, b_i, x_i, and y_i $(1 \le a_i, b_i \le n, a_i \ne b_i, 1 \le x_i, y_i \le 10^5)$, representing a transport line operated by the company.

It is guaranteed that any two stations can be reached from each other through the railway network.

Output

Output a single integer on a line, representing the maximum daily revenue.

Example

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

Problem C. graph

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

Given an undirected graph without self-loops, your task is to assign either 0 or 1 to each node. Subsequently, any edge connecting two nodes with different values will be removed. Each **connected component** should then have at least one **Eulerian path**. If there are multiple ways to assign values to the nodes, you can choose any one of them.

Eulerian path: an **Eulerian trail** (or **Eulerian path**) is a trail in a finite graph that visits every edge exactly once (allowing for revisiting vertices).

Connected component: a **component** of an undirected graph is a connected subgraph that is not part of any larger connected subgraph.

Input

The first line contains two integers n, m $(1 \le n \le 100, 0 \le m \le 10000)$, representing the number of nodes and edges in the graph, respectively.

For the next m lines, each contains two integers u_i, v_i $(1 \le u_i, v_i \le n, u_i \ne v_i)$, which means there is an edge connecting node u_i and v_i. It is possible to have multiple edges between the same pair of nodes.

Output

If it is possible to assign the values, print n values of 0 or 1 in a single line. The i-th of them represents the value assigned to the i-th node. Otherwise, print -1.

If there are multiple solutions, you can print any of them.

Example

standard input	standard output
3 6 2 3 1 3 2 1 2 1 1 2 1 3	0 1 0

Problem D. The Game of Eating

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

Today, Fallleaves01 and his $n-1$ friends went out for dinner. They intend to order a total of k dishes. To ensure everyone gets their favorite dish, they decided to take turns ordering one dish each in sequential order and repeat until they have ordered a total of k dishes.

Please be aware that **the dishes ordered by one person can be shared and tasted by everyone**. To ensure a wide variety of flavors and experiences, it is important for individuals to **avoid ordering the same dish multiple times**.

There are a total of m dishes available for selection. Everyone knows each person's preferences for every dish. This information is represented by an $n \times m$ matrix called A, where $A_{i,j}$ denotes the degree to which the $i-th$ person likes the $j-th$ dish.

Assume that each person only cares about their own enjoyment of the dishes without considering others. In other words,if they ultimately choose dishes $p_1, p_2..., p_k$, person i aims to maximize $\sum_{1 \le j \le k} A_{i,p_j}$.

We also found that for every different $1 \le x, y \le m, A_{i,x} \ne A_{i,y}$.

Fallleaves01 is curious to know what dishes will be on the table if everyone makes optimal decisions.

Input

The input consists of multiple test cases. The first line of each test case contains an integer t, indicating the number of test cases. The following lines describe each test case.

For each test case, the first line contains three integers n, m, and k $(1 \le n \le 2000, 1 \le k \le m \le 2000)$. These represent the number of people, the number of dishes, and the number of dishes planned to be ordered, respectively.

The next n lines contain m integers per line. The j-th number on the i-th line represents the degree to which the i-th person likes the j-th dish, denoted as $A_{i,j}$ $(1 \le A_{i,j} \le 10^9)$.

It is guaranteed that the sum of n and the sum of m do not exceed 2000.

Output

For each test case, output a row of k integers in **increasing** order, representing the final selected dishes.

Example

standard input
3 3 4 2 3 2 1 4 3 1 2 4 1 2 3 4 3 4 3 1 2 3 4 3 1 2 4 1 3 2 4 3 20 10 12 25 24 2 23 1 7 17 10 13 9 4 30 29 11 20 14 27 19 18 9 1 22 26 15 16 11 29 18 24 3 21 25 6 19 7 14 13 17 28 26 27 16 2 17 20 12 4 3 1 24 19 9 28 8 18 15 29 13 22

standard output
1 4 1 3 4 1 2 3 4 5 8 13 14 18 20

Problem E. Square

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

Find a number $0 \leq y \leq 10^9$ so that the square of y starts with x in Decimal.

Formally, given a integer x, find an integer $y (0 \leq y \leq 10^9)$ such that there exists a nonnegative integer k that satisfies $\lfloor \frac{y^2}{10^k} \rfloor = x$.

Input

Each test contains multiple test cases. The first line of input contains a single integer $t (1 \leq t \leq 10^5)$ — the number of test cases.

Each test case contains an integer $x (0 \leq x \leq 10^9)$ - the x described in the problem statement.

Output

For each test case, output an integer $y (0 \leq y \leq 10^9)$.If there are multiple results, you can output any one. If no y satisfies the condition, output -1.

Example

standard input	standard output
3 1 123 781273981	4 111 -1

Problem F. Link with Chess Game

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

Given an undirected graph with n vertices and $n-1$ edges. The i-th edge connects vertex i and vertex $i+1$.

There are three chess pieces: red, green, and blue. The red piece is located at vertex r, the green piece is located at vertex g, and the blue piece is located at vertex b. (Different pieces may occupy the same vertex.)

Alice and Bob take turns performing the following operation, with Alice going first:

- Select a chess piece and move it to an adjacent vertex (different pieces may occupy the same vertex).

If, after a move, the ordered triple (r, g, b) formed by the positions of the three pieces has appeared at some previous moment, the player who made that move loses, and the other player wins. (The initial state is considered to have appeared before.)

Assuming both players adopt an optimal strategy, who will win the game?

Input

Input consists of multiple test cases.

The first line contains an integer T ($1 \le T \le 10^4$), indicating the number of test cases.

Following that are T lines, each containing four positive integers n, r_0, g_0, b_0 ($2 \le n \le 10^5, 1 \le r_0, g_0, b_0 \le n$). These represent a query, where r_0, g_0, b_0 are the initial positions of the red, green and blue chess pieces, respectively.

It is guaranteed that $\sum n \le 10^6$.

Output

For each test case, if Alice wins, please output 'Alice'; otherwise, please output 'Bob'.

Example

standard input	standard output
2 2 2 2 2 3 3 3 3	Alice Bob

Problem G. Link with Centrally Symmetric Strings

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

Given a string, determine if it can be represented as the concatenation of several centrally symmetric substrings.

Formally, a string S is centrally symmetric if and only if it satisfies one of the following conditions:

- S is an empty string.
- $S = o|s|x|z$, i.e., S is one of the four letters: o, s, x and z.
- $S = bSq|dSp|pSd|qSb|nSu|uSn|oSo|sSs|xSx|zSz$, i.e., S starts and ends with a pair of centrally symmetric letters, and the middle part is also a centrally symmetric substring.

A string S is considered good if and only if there exists an integer $n \geq 1$ such that $S = T_1 T_2 \cdots T_n$, where T_i $(1 \leq i \leq n)$ is a centrally symmetric substring.

Given a string S, determine whether it is a good string.

Input

The input contains multiple test cases.

The first line contains an integer $T (1 \leq T \leq 10^6)$, indicating the number of test cases.

The following T lines each contain a string S $(1 \leq |S| \leq 10^6)$ consisting of lowercase English letters, representing the string to be tested.

It is guaranteed that $\sum |S| \leq 5 \times 10^6$, where $|S|$ represents the length of string S.

Output

For each test case, if the string is good, output "Yes"; otherwise, output "No".

You can output each letter in any case (lowercase or uppercase). For example, the strings "yEs", "yes", "Yes", and "YES" will be accepted as a positive answer.

Example

standard input	standard output
3 sosos hahaha sbzzq	Yes No Yes

Note

For the third test case, the string can be divided into two substrings, ‘s’ and ‘bzzq’, both of which are centrally symmetric substrings. Therefore, the output is ‘Yes’.

Problem H. 0 and 1 in BIT

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

There are many 0 and 1 in both bit and BIT.

In BIT, the definition of 0 and 1 is also somehow ambiguous. 0 can sometimes become 1 and vice versa.

Now, given an event string containing only A and B with length n, you're required to answer Q questions (l_{real}, r_{real}, x) which ask what will binary string x be after experiencing the events in interval $[l_{real}, r_{real}]$ (The leftmost bit is the most significant bit for x).

Here's the illustration for the event A, B.

A: change 0 in x to 1 and change 1 in x to 0 simultaneously.

B: execute $x = x + 1$ where x is viewed as an integer in binary representation. Specially, if x contains only 1, x will become a binary string with all 0.

You are required to answer each problem online. For this propose, if we ask (l, r) in the $i-$th query, it means the actually asked segment is defined as:

$$l_{real} = min((ans_{i-1} \oplus l)\%n + 1,\ (ans_{i-1} \oplus r)\%n + 1)$$

$$r_{real} = max((ans_{i-1} \oplus l)\%n + 1,\ (ans_{i-1} \oplus r)\%n + 1)$$

where $\oplus$ is the exclusive OR operation and ans_{i-1} is the answer for the $(i-1)-$th query (viewed as an integer in binary representation) with $ans_0 = 0$.

Input

The first line contains two integers, $n(3 \le n \le 2 \times 10^5), q(1 \le q \le 2 \times 10^5)$, the length of event string and the number of queries.

The second line contains the event string $S(|S| = n)$ and S contains only A and B.

In the following q lines, each line contains a query $l, r, x(1 \le l \le r \le n, 1 \le |x| \le 50)$, and x is a binary string containing only 0 and 1.

Output

For each query, output the binary representation of x after the events in interval $[l, r]$.

Example

standard input	standard output
10 3 BAABABABBA 1 8 0001 3 5 110 6 10 0101010101	0011 111 0101010110

Note

The actually asked segments in the samples are $(2,9),(1,7),(2,4)$, respectively.

Problem I. Link with Gomoku

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

"The more you win, the stronger your confidence becomes. The stronger your confidence, the more arrogant you become. The more arrogant you are, the easier it is to make mistakes. The more mistakes you make, the more you lose. So the more you win, the more you lose."

Link and Fall are playing Gomoku on a board with n rows and m columns, but neither of them wants to win (because the more they win, the more they lose). Therefore, they have decided to achieve a draw. Your task is to design a strategy for them to place their pieces on the board in such a way that the game ends in a draw.

If you have never played Gomoku before, here are the rules:

Black plays the first move (represented by 'x'), and white plays the second move (represented by 'o'). Each player takes turns placing their piece on an empty spot on the board. The game ends immediately if either player has five consecutive pieces in a row, column, or diagonal (both positive and negative). If the board is filled but neither player has five consecutive pieces, the game is a draw.

In this problem, the forbidden move rule is not used.

Input

The input contains multiple test cases.

The first line contains an integer $T(1 \leq T \leq 10^3)$, representing the number of test cases.

Next, there are T lines, each containing two integers n and m $(1 \leq n, m \leq 10^3)$, representing a test case.

It is guaranteed that $\sum n \times m \leq 10^6$.

Output

For each test case, you need to output n lines, each containing m characters, representing the final state of the board. The j-th character of the i-th line $c_{i,j}$ represents the piece at row i, column j of the board. If $c_{i,j}$ is 'x', it means the cell at row i, column j contains a black piece. If $c_{i,j}$ is 'o', it means the cell at row i, column j contains a white piece. If $c_{i,j}$ is any other character, you will receive a 'Wrong Answer' result.

If there is a line of five consecutive pieces on the board or **the state you output cannot be the final state** of a Gomoku game according to the aforementioned rules, you will receive a 'Wrong Answer' result.

It can be shown that at least one solution exists. If there are multiple solutions, you can output any of them.

Example

standard input	standard output
2 4 4 5 5	oooo xxxx oooo xxxx oooox xxxxo oooox xxxxo xoxox

Problem J. Smoke

Input file:	`standard input`
Output file:	`standard output`
Time limit:	8 seconds
Memory limit:	512 megabytes

You are right, but "Smoke God" is a brand new open-world adventure game independently developed by Ding Zhenzhen. The game takes place in a fantasy world called "Litang", where people chosen by the gods will be granted "electronic cigarettes" to guide the power of nicotine. You will play a mysterious character named "Furong Wang" and meet animal friends with different personalities and unique abilities during your free travels. Together, you will defeat powerful enemies, find lost family members, and gradually uncover the truth about "Litang".

The harm of electronic cigarettes to the body is significant, and the damage to body depends on the level of addiction.

Let's assume that Ding Zhenzhen's current addiction level is j, and the damage to body is x. When she smokes i-th electronic cigarette:

- There is a probability of p that the addiction level increases by one, and x is multiplied by A_j.
- There is a probability of $1 - p$ that the addiction level remains unchanged, and x is multiplied by $B_j + C_i$.

However, you do not know Ding Zhenzhen's current addiction level and the current damage to her body. You only know that if her addiction level is j, the current damage to her body is D_j.

Assuming that the initial addiction level to Ding Zhenzhen's body is equally likely to be any integer from 1 to m. Given N, what is the expected damage to her body after smoking $n = 1, 2 \cdots, N$ cigarettes?

Input

The first line contains two integers, N, m and p ($1 \le N, m \le 10^5, 0 \le p < 998244353$), representing the number of smoking, the maximum addiction level and the probability to increase the addiction level when smoking an electronic cigarette respectively.

The second line contains $m + N - 1$ integers, where the i-th integer A_i ($0 \le A_i < 998244353$) represents the damage coefficient when the addiction level increases from i to $i + 1$ after smoking an electronic cigarette.

The thrid line contains $m+N-1$ integers, where the i-th integer B_i ($0 \le B_i < 998244353$) represents one of the damage coefficient when the addiction level remains at i after smoking an electronic cigarette.

The fourth line contains N integers, where the i-th integer C_i ($0 \le C_i < 998244353$) represents one of the damage coefficient after smoking i-th electronic cigarette.

The fifth line contains m integers, where the i-th integer D_i ($0 \le D_i < 998244353$) represents the initial damage to her body if the initial addiction level was i.

Output

Output N integers for $n = 1, 2, \cdots, N$ representing expected damage to her body after smoking n cigarettes, module 998244353.

Examples

standard input	standard output
1 1 499122177 2 2 2 1	3
2 2 546658423 260257767 682179463 892187612 142884587 872658039 89862243 117086929 104310686 342803717 47992235	828045057 157615404

Problem K. Box

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

That's enough. There's no need to bring Genshin Impact into every little matter. Genshin Impact hasn't done anything to offend or harm you. Why do you keep mindlessly bashing it? miHoYo puts a lot of effort into promoting Chinese culture through their games, and all you do is sit behind a keyboard and criticize a conscientious company. People like you are ruining the future of Chinese games.

Now, the Traveler has n boxes in their possession, some of which have lids that can be shifted left or right by at most one step. If the i-th box is covered with a lid, you gain a_i score. Determine the maximum total game score achievable by covering boxes with lids after shifting serveral(possibly, zero) lids.

Input

The first line contains a integer, $n(1 \le n \le 10^6)$, representing the number of boxes.

The second line contains n integers, where the i-th integer a_i $(0 \le a_i \le 10^9)$ represents the value of i-th box.

The third line contains n integers, where the i-th integer b_i $(0 \le b_i \le 1)$ represents whether i-box has a lid. $b_i = 0$ means there is no lid on the box.

Output

An integer represents the answer.

Examples

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

Problem L. Link doesn't want to cut tree

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

Link has a convex quadrilateral farm, and he planted four trees at the four vertices of the farm to mark it.

Now Link wants to expand his farm, and he has some requirements for the new farm:

- The new farm should be a diamond shape, which is a quadrilateral with equal sides.
- The new farm area should be twice that of the original farm.
- Link doesn't want to cut trees, so the four trees should be placed on each side of the new farm.

Formally, given a convex quadrilateral $ABCD$, you should calculate four distinct points $A'B'C'D'$, which satisfies:

- $A'B'C'D'$ is a diamond shape, that is, $|A'B'| = |B'C'| = |C'D'| = |D'A'|$.
- $S_{A'B'C'D'} = 2S_{ABCD}$.
- Point A is located on segment $A'B'$. Point B is located on segment $B'C'$. Point C is located on segment $C'D'$. Point D is located on segment $D'A'$.

You need to calculate $A'B'C'D'$ or indicate that this is impossible.

Input

Each test contains multiple test cases. The first line of input contains a single integer $t (1 \le t \le 10^4)$ — the number of test cases.

Each test case contains 8 integers $x_A, y_A, x_B, y_B, x_C, y_C, x_D, y_D$ — the coordinates of points A, B, C, D. The absolute value of all integers should not exceed 10^4

Output

For each test case, if there is at least a solution, print "Link doesn't cut tree!", then print 8 numbers, representing $x_{A'}, y_{A'}, x_{B'}, y_{B'}, x_{C'}, y_{C'}, x_{D'}, y_{D'}$ respectively.

If there is no possible solution, print "Link cut tree!".

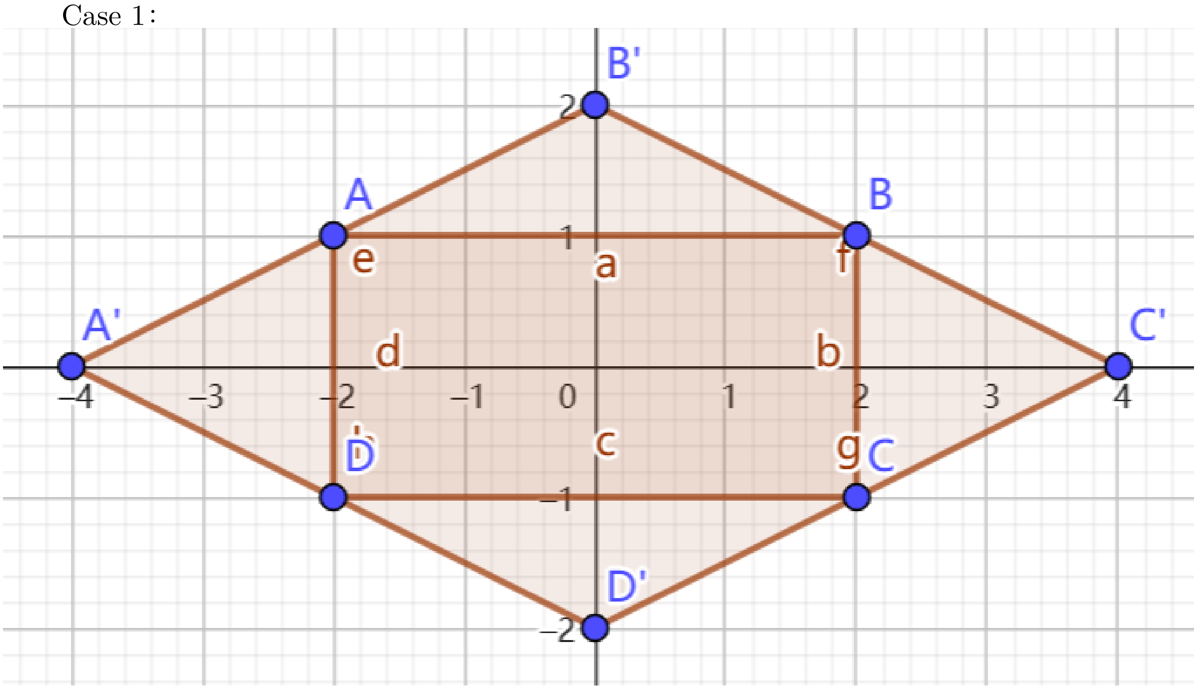
The answer would be considered correct if the absolute error of all conditions is no more than 10^{-5}.

Example

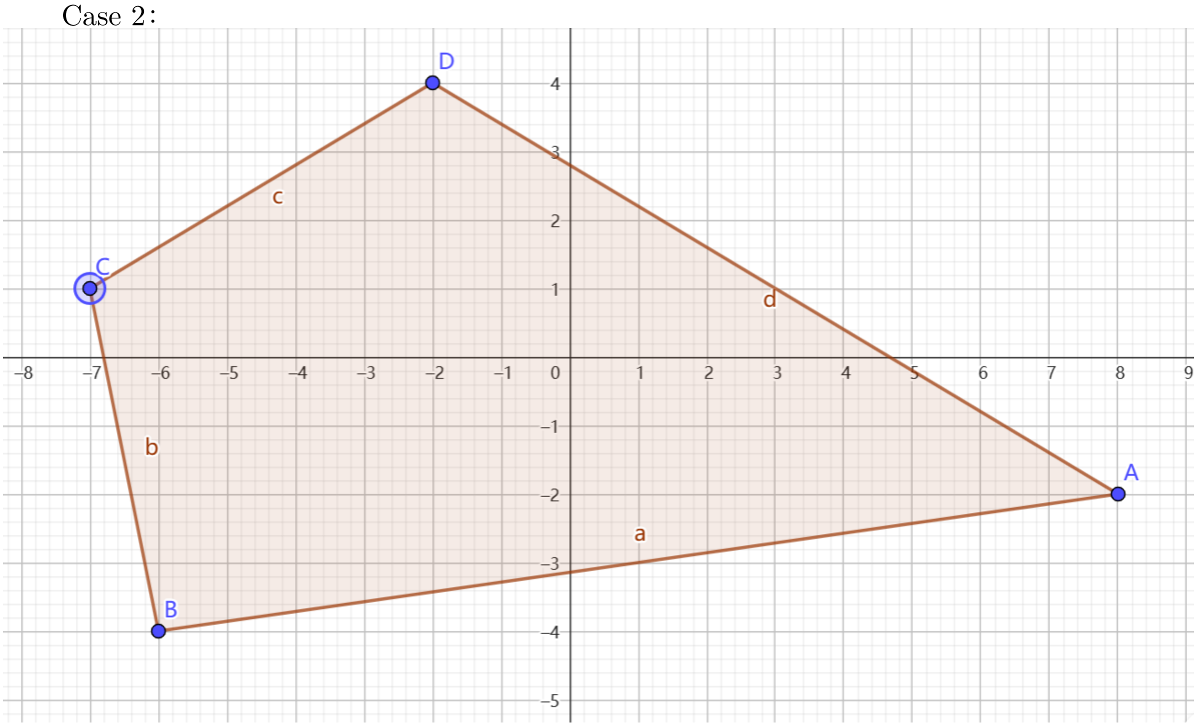
standard input	standard output
3 -2 1 2 1 2 -1 -2 -1 8 -2 -6 -4 -7 1 -2 4 -4 4 4 4 8 -2 -6 -3	Link doesn't cut tree! -4 0 0 2 4 0 0 -2 Link cut tree! Link doesn't cut tree! -12 0 0 6 12 0 0 -6

Note

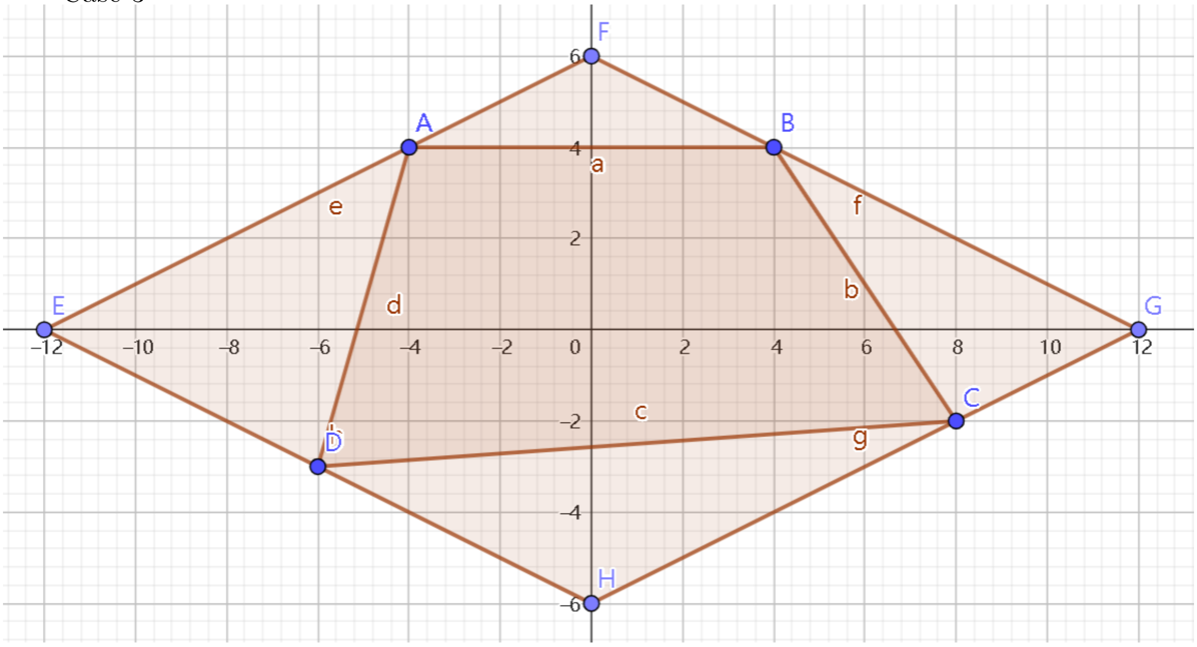
Case 1:



Case 2:



Case 3:

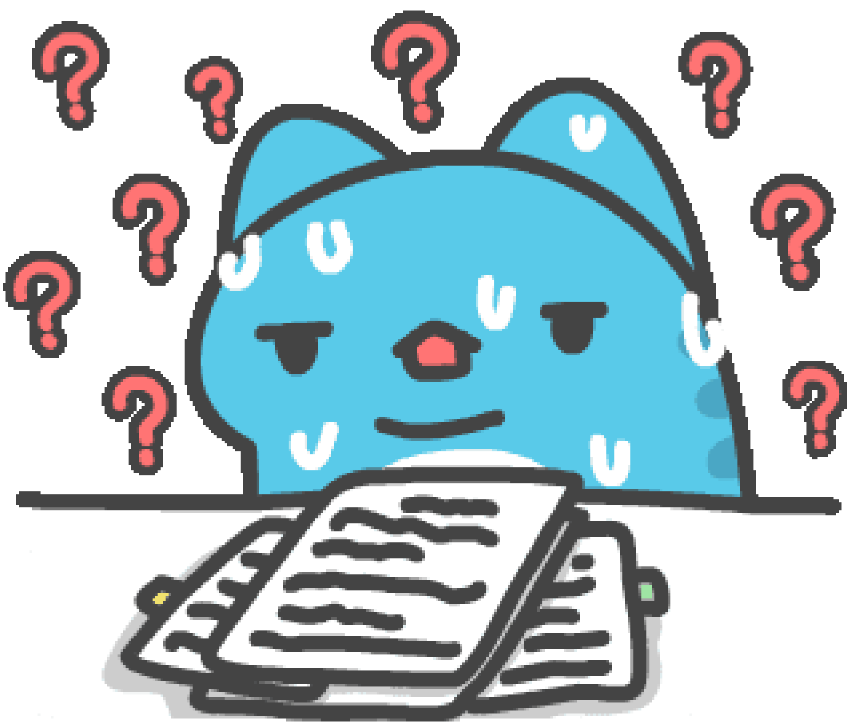


Problem M. Fundamental Skills in Data Structures

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	256 megabytes

"To strengthen the fundamental skills in data structures and gain a profound understanding of the essence of data structures, as well as to implement code effectively, we need to make efforts to study the profound connotation of data structures..."

The ACM Training Base is having a meeting, but after listening to only two sentences, Capoo the cat falls asleep. When he wakes up, he feels at a loss facing the homework left behind. Luckily, he knows a data structure expert, which is you! In order to prevent poor Capoo from being scolded by his senior, you decide to help him complete his data structure homework.



The homework questions are as follows:

Given a tree rooted at node 1, each node in the tree has a node weight $a_i \in \{0, 1\}$. Now, you need to perform q operations, which can be divided into two types:

- $1\ u\ v\ x$, which means setting the node weights of all the nodes on the simple path between u and v (inclusive) to x.
- $2\ u$, which means querying the number of pairs of nodes (x, y) in the subtree rooted at u such that $x < y$ and $a_x \oplus a_y \oplus a_{lca(x,y)} = 0$ holds.

Here, $\oplus$ represents the bitwise XOR operation, similar to the XOR operator (^) in C/C++.

$lca(x, y)$ refers to the lowest common ancestor of nodes x and y. For example, in Sample 3, $lca(4, 5) = 3$.

Input

The first line consists of two integers, n and q ($2 \le n \le 3 \times 10^5, 1 \le q \le 3 \times 10^5$), representing the number of nodes in the tree and the number of operations, respectively.

The second line contains n integers, a_i ($a_i \in \{0, 1\}$), representing the initial node weights.

The third line contains $n-1$ integers, where the ith integer, f_i ($1 \le f_i \le i$), represents the existence of an edge between the $(i+1)$th node and the f_ith node.

Following are q lines, each representing an operation:

For each operation, the first integer of the line, op ($op \in \{1, 2\}$), denotes the type of operation.

- If $op = 1$, then the next three integers, u, v, and x ($1 \le u, v \le n, x \in \{0, 1\}$), represent the two endpoints of the covered chain and the assigned weight for that chain.
- If $op = 2$, then the next integer, u ($1 \le u \le n$), represents the root node of the subtree for which a query is required.

Output

For each query, output a single line containing an integer representing the answer to that query.

Examples

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

Note

For Sample 3, in the first query, after the first two coverings, the node weights of the tree become $\{0, 0, 0, 0, 1\}$, and the valid point pairs in subtree 3 are $(3, 4)$.

In the second query, after the first three coverings, the node weights of the tree become $\{1, 1, 0, 0, 1\}$, and the valid point pairs in subtree 1 are $(1, 3), (1, 4), (2, 3), (2, 4), (3, 4)$.