Problem A. Xcellent Tree Query Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

Given a tree consisting of n nodes. Each node initially have a color c_i.

You are given m commands, each of them is one of the follows:

- $1\ x$: Cut the x-th edge of the input.
- $2\ x\ l\ r\ v$: For every y currently connected with x (that is, no edges lying on the simple path from x to y is cut), if $l \le c_y \le r$, set c_y to v.
- $3\ x\ l\ r$: Count the number of y currently connected with x, that satisfy $l \le c_y \le r$.

Input

The first line of the input contains two integers n ($1 \le n \le 10^5$) and m ($1 \le m \le 5 \times 10^5$).

The second line contains n integers $c_1, c_2, \ldots, c_n$ ($1 \le c_i \le n$).

Each of the following $n-1$ lines contains two integers u, v ($1 \le u, v \le n$, $u \ne v$) - an edge of the tree.

Each of the following m lines contains one of the three commands listed above.

It's guaranteed that no edges is cut twice or more.

It's also guaranteed that in command of type 2 and 3, $1 \le l \le r \le n$, $1 \le x, v \le n$.

Output

For every command of type 3, output the answer in a single line.

Examples

standard input	standard output
10 19 1 2 3 4 5 6 7 8 9 10 1 2 2 3 3 4 4 5 4 6 5 7 5 8 1 9 9 10 3 4 1 10 3 8 3 4 2 9 9 10 1 3 4 1 10 3 2 8 10 1 4 3 8 1 10 3 7 7 8 2 5 5 7 9 3 7 9 10 2 1 1 2 3 3 1 3 3 1 1 3 1 3 3 2 10 3 9 4 3 1 4 4 2 3 3 5 9 3 3 6 10 3 3 6 6	10 2 10 1 3 2 2 5 3 3 4 1

Problem B. Random Nim Game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Nim is a two-player mathematic game of strategy in which players take turns removing stones from distinct heaps. On each turn, a player must remove at least one stone, and may remove any number of stones provided they all come from the same heap. The person who makes the last move (i.e., who takes the last stone) wins.

Alice and Bob is tired of playing Nim under the standard rule, so they want to play Nim randomly. On each turn, a player must select any one of the heaps. Assuming the heap he selects contains x stones, he will randomly choose a integer number y from $[1, x]$, and remove y stone(s) from the heap. Note that the selected heap can be arbitrary.

Alice will play first. Calculate the probability of Alice winning, modulo 998244353.

Input

The input consists of multiple test cases. The first line contains a single integer T $(1 \le T \le 500)$ - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n $(1 \le n \le 10^5)$ - the number of heap(s).

The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$ - the number of stone(s) of each heap.

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

Output

For each test case, print a single integer - the probability of Alice winning, modulo 998244353.

Examples

standard input	standard output
2 2 1 1 1 2	0 499122177

Problem C. Rotation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

There is a binary tree with the root at vertex 1, and each vertex u in the tree has a weight a_u. You can perform the following two operations:

1. If vertex u has left child v and right child w, and its left child v has a right child x, then you can perform a "rotation" on these four vertices. Let $a'_u = a_v, a'_w = a_u, a'_x = a_w, a'_v = a_x$, and then let $a_u = a'_u, a_v = a'_v, a_w = a'_w, a_x = a'_x$.
2. If vertex u has left child v and right child w, and its right child w has a left child x, then you can perform a "rotation" on these four vertices. Let $a'_u = a_v, a'_w = a_u, a'_x = a_w, a'_v = a_x$, and then let $a_u = a'_u, a_v = a'_v, a_w = a'_w, a_x = a'_x$.

Find the number of different point weight arrays a that can be obtained by performing a series of operations, modulo 998244353.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 1000$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n ($1 \le n \le 10^5$) - the number of vertices in the tree.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$) - the point weights.

Each of the following n lines contains two integers l_i, r_i ($0 \le l_i, r_i \le n$) - the left child and right child of vertex i. If a child does not exist, it is represented by 0.

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

Output

For each test case, print a single integer - the number of different point weight arrays a that can be obtained by performing a series of operations, modulo 998244353.

Examples

standard input	standard output
2 4 1 2 2 1 2 3 0 4 0 0 0 0 8 1 2 3 4 5 6 7 8 2 3 4 5 6 7 0 0 0 0 0 8 0 0 0 0	2 5040

Problem D. Medians Strike Back

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Define the **median** of a sequence of length n as:

If n is odd, the **median** is the number ranked $\lfloor\frac{n+1}{2}\rfloor$ if we sort the sequence in ascending order.

If n is even, the **median** is the number that has more occurences between the numbers ranked $\lfloor\frac{n}{2}\rfloor$ and $\lfloor\frac{n}{2}+1\rfloor$ if we sort the sequence in ascending order. If they appeared for the same number of times the smaller one is the **median**.

Define the **shikness** of a sequence A as the number of occurences of the **median** of A.

Define the **nitness** of a sequence A as the maximum **shikness** over all continuous subsequences of A.

You want to find a sequence A of length n, satisfying $1 \leq A_i \leq 3$ for every $1 \leq i \leq n$, with the minimum **nitness**.

Calculate the **nitness** of such sequence.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \leq T \leq 2 \times 10^5$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n ($1 \leq n \leq 10^9$).

Output

For each test case, print a single integer - the **nitness** of such sequence.

Examples

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

Problem E. Subsequence Not Substring

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3.5 seconds
Memory limit:	512 megabytes

Given a string S consisting of only lowercase latin letters。

Find the lexicographic smallest string T, satisfying T is a subsequence of S, but T is not a substring of S, or determine such T doesn't exist.

Input

The input consists of multiple test cases. The first line contains a single integer T $(1 \leq T \leq 2 \times 10^5)$ - the number of test cases. Description of the test cases follows.

The first line of each test case contains one string S $(1 \leq |S| \leq 10^5)$.

It's guaranteed that $\sum |S| \leq 5 \times 10^6$.

Output

For each test case, print a single string T, satisfying T is a subsequence of S, but T is not a substring of S. If such T doesn't exist, print `-1`.

It's guaranteed that $\sum |T| \leq 2 \times 10^6$.

Examples

standard input	standard output
4 bbacb aaabaaba gugguggguggggu zzzzzzzzzzzzzzzz	ab aaaa ggggg -1

Problem F. Product of Sorting Powers

Input file:	`standard input`
Output file:	`standard output`
Time limit:	11.5 seconds
Memory limit:	64 megabytes

Given a sequence A of length n.

There are m queries, each given l, r, let $B_1, B_2, \ldots, B_{r-l+1}$ as the result of sorting $A_l, A_{l+1}, \ldots, A_r$, calculate:

$$\left(\prod_{i=1}^{r-l} {B_i}^{B_{i+1}}\right) \bmod (10^9 + 7)$$

Input

The first line of the input contains two integers n, m ($1 \le n \le 10^6$, $1 \le m \le 5000$) - the length of A and the number of queries.

The second line contains n integers $A_1, A_2, \ldots, A_n$ ($1 \le A_i \le 10^9$).

Each of the following m lines contains two integers l, r ($1 \le l \le r \le n$).

Output

For each query, print a single integer - the answer of the query, modulo $10^9 + 7$.

Examples

standard input	standard output
6 5 8 2 5 9 7 7 1 4 2 6 1 3 3 6 2 5	588255953 426219106 12500000 575819351 16793812

Problem G. Sum of Binomial Coefficients

Input file:	`standard input`
Output file:	`standard output`
Time limit:	14.5 seconds
Memory limit:	512 megabytes

Given an **increasing** sequence A of length m.

There are q queries, each given N, R, calculate:

$$\left(\sum_{i=1}^{R}\binom{N}{A_i}\right) \bmod 998244353$$

Input

The first line of the input contains two integers m, q $(1 \le m, q < 2^{17})$ - the length of A and the number of queries.

The second line contains m integers $A_1, A_2, \ldots, A_m$ $(1 \le A_1 < A_2 < \cdots < A_m < 2^{17})$.

Each of the following q lines contains two integers N, R $(1 \le N \le 10^9,\ 1 \le R \le m)$.

Output

For each query, print a single integer - the answer of the query, modulo 998244353.

Examples

standard input	standard output
6 10 2 4 5 6 8 10 5 2 5 1 4 3 8 5 5 2 5 1 10 6 7 6 6 2 8 6	15 10 7 183 15 10 763 84 30 183

Notes

There is a template for polynomial operations in https://paste.ubuntu.com/p/ppX3GhvKYN/ or https://www.luogu.com.cn/paste/g8nbrg2u.

Problem H. HEX-A-GONE Trails

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 megabytes

Consider a tree of n nodes. Two OPs, OP I and OP II are playing a game on the tree. In the beginning, OP I and OP II are at node x and node y, respectively. Then they take turns to move, OP I moves first.

In each move, a player at node i must choose a neighboring node j and move to j. Remind that a player is not allowed to move to the other player's current position. After this move, node i becomes invalid, meaning it cannot be moved to in the following moves of both players.

If a player cannot make a valid move, he will lose the game.

Please determine whether OP I has a strategy to make sure he will win.

Input

The input consists of multiple test cases. The first line contains a single integer T $(1 \le T \le 500)$ - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n $(1 \le n \le 10^5)$.

The second line contains two integers x, y $(1 \le x, y \le n,\ x \ne y)$.

Each of the following $n-1$ lines contains two integers u, v $(1 \le u, v \le n,\ u \ne v)$ - an edge between u, v.

It's guaranteed that $\sum n \le 6 \times 10^5$.

Output

For each test case, print a single integer - If OP I has a strategy to make sure he will win, output 1. Otherwise output 0.

Examples

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

Problem I. Colorings Counting

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Given a ring of n nodes. It's required to color each node with one of the colors in $0 \sim m-1$, and ensure that adjacent points have different colors.

Consider two colorings are different if and only if two colorings sequences a, b cannot be transformed into each other through several of the following three operations:

- Forall $1 \le i \le n$, $a'[i] \leftarrow a[i \bmod n + 1]$, then $a[i] \leftarrow a'[i]$;
- Forall $1 \le i \le n$, $a'[i] \leftarrow a[n - i + 1]$, then $a[i] \leftarrow a'[i]$;
- Forall $1 \le i \le n$, $a'[i] \leftarrow (a[i] + 1) \bmod m$, then $a[i] \leftarrow a'[i]$.

Calculate the different colorings, modulo 998244353.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 100$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains two integer n, m ($4 \le n \le 10^{18}$, $2 \le m \le 10^{18}$).

It's guaranteed that n, m are not multiples of 998244353.

It's guaranteed that there will be no more than 40 test cases with $n, m > 20$.

It's guaranteed that there will be no more than 20 test cases with $n, m > 10^5$.

It's guaranteed that there will be no more than 5 test cases with $n, m > 10^{13}$.

Output

For each test case, print a single integer - the different colorings, modulo 998244353.

Examples

standard input	standard output
4 4 2 4 3 6 3 2023 808	1 2 5 304535191

Problem J. Widely Known Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	20 seconds
Memory limit:	512 megabytes

Given a string s of length n consisting of lowercase English letters.

You are given m patterns, where the i-th pattern is $s[l_i \dots r_i]$ (indices of s start from 1).

Now, there are q queries, and each query provides $[L_i, R_i]$. For each query, you need to find how many j that the j-th pattern occurs in $s[L_i \dots R_i]$.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 10$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains three integer n, m, q ($1 \le n \le 5 \times 10^5$, $1 \le m, q \le 10^6$).

The second line contains a string s of length n.

Each of the following m lines contains two integers l_i, r_i ($1 \le l_i \le r_i \le n$).

Each of the following q lines contains two integers L_i, R_i ($1 \le L_i \le R_i \le n$).

It's guaranteed that $\sum n \le 9 \times 10^5$, $\sum m, \sum q \le 2 \times 10^6$.

Output

For each query, print one integer - the number of patterns occur in the given substring.

Examples

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

Problem K. Three Operations

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1.5 seconds
Memory limit:	512 megabytes

Given three integers x, a, b. You can do the following three operations several times:

- set x to $x - 1$;
- set x to $\lfloor \frac{x+a}{2} \rfloor$;
- set x to $\lfloor \sqrt{x+b} \rfloor$.

Calculate the smallest number of operations to set x to 0.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 2 \times 10^4$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains three integers x, a, b ($0 \le x, a, b \le 10^{18}$).

Output

For each test case, print one integer - the smallest number of operations to set x to 0.

Examples

standard input	standard output
5 1 1 4 5 1 4 19 1 9 8 1 0 1145141919810 114514 1919810	1 4 5 3 1389

Problem L. Landmine

Input file:	`standard input`
Output file:	`standard output`
Time limit:	12 seconds
Memory limit:	512 megabytes

Given a tree where each edge has a length. Each node contains a landmine, and the i-th node's landmine has an explosion radius of r_i.

We define $dist(i, j)$ as the shortest distance between vertex i and vertex j in the tree. In other words, $dist(i, j)$ is the sum of edge lengths along the unique simple path between vertex i and vertex j.

When the landmine at vertex i explodes, after one second, all landmines within distance r_i from vertex i will explode together. In other words, for all landmines j satisfying $dist(i, j) \le r_i$, if they have not exploded yet, they will be detonated one second after the landmine at vertex i explodes.

For each $i = 2, 3, \ldots, n$, you want to calculate how long it will take for the first landmine at vertex 1 to be detonated after detonating the landmine at vertex i, or if it can never be detonated.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 100$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n ($1 \le n \le 10^5$).

The second line contains $n - 1$ integers $r_2, r_3, \ldots, r_n$ ($0 \le r_i \le 10^{18}$).

Each of the following $n - 1$ lines contains three integers u, v, w ($1 \le u, v \le n$, $u \ne v$, $1 \le w_i \le 10^{12}$) - an edge between u and v with a length of w.

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

Output

For each test case, print $n - 1$ integers - the time in seconds after detonating the landmine at vertex i, at which the first landmine at vertex 1 will be detonated. If it can never be detonated, output -1.

Examples

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

Problem M. Minimal and Maximal XOR Sum

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Given a permutation $p_1, p_2, \ldots, p_n$ of $1 \sim n$. You can perform several operations.

In each operation you can choose an interval $[l, r]$ and reverse the elements $p_l, p_{l+1}, \ldots, p_r$ to $p_r, p_{r-1}, \ldots, p_l$, the weight of this operation is $r - l + 1$.

You can perform any number of operations, and your goal is to make $p_i = i$ at last.

Please calculate the minimum and maximal XOR sum of the weight of all the operations.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \le T \le 2 \times 10^5$) - the number of test cases. Description of the test cases follows.

The first line of each test case contains one integer n ($1 \le n \le 10^5$).

The second line contains n integers $p_1, p_2, \ldots, p_n$.

It's guaranteed that $\sum n \le 6 \times 10^5$.

Output

For each test case, print two integers - the minimum and maximal XOR sum of the weight of all the operations.

Examples

standard input	standard output
3 3 1 3 2 3 3 1 2 6 1 2 5 6 3 4	2 3 0 1 0 5