Problem A. Alice and Her Lost Cat

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

Alice wants to find her lost cat in the park.

The park is a rooted tree consisting of n vertices. The vertices are numbered from 1 to n, and the root is vertex 1.

Alice is now at vertex 1. She knows that the cat has run from vertex 1 to some leaf of the tree, and no vertex is visited more than once. A leaf is a vertex without children.

There is a monitor on each vertex. The monitor on vertex i can observe whether cat has visited vertex i and which vertex the cat has gone to (if vertex i is not a leaf). It takes a_i seconds for Alice to check the data of the i-th monitor.

Alice can also search some leaves by herself. It takes t_i seconds to search i leaves. Note that i is the count of vertices instead of the label of a vertex.

Help Alice to determine which monitors to be checked and which leaves to be searched so that the location of the cat can be uniquely determined, and the total time needed is minimum possible. Note that the monitors to be checked and the leaves to be searched should be decided at the beginning, and should not change after that.

Find the minimum time.

Input

Each test contains multiple test cases.

The first line contains one single integer T $(1 \le T \le 10)$ denoting the number of test cases. The description of the T test cases follows.

The first line of each test case contains an integer n $(1 \le n \le 2000)$ — the size of the tree.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$.

The third line of each test case contains n integers $t_1, t_2, \ldots, t_n$ $(1 \le t_i \le 10^9, t_i \le t_{i+1})$.

Then $n-1$ lines follow, each containing two integers x and y $(1 \le x, y \le n, x \ne y)$ denoting an edge connecting vertex x with vertex y. It is guaranteed that these edges form a tree.

Output

For each test case, output an integer in one line — the minimum time needed to determine the cat's location.

Example

standard input	standard output

2
8
4 2 5 2 4 2 3 2
5 5 6 7 8 9 10 13
1 2
2 3
1 4
1 5
4 6
3 7
5 8
8
4 2 3 3 2 4 4 3
4 6 8 8 9 9 14 17
1 2
2 3
3 4
3 5
4 6
3 7
3 8
	4
	3

Problem B. Ayano and sequences

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

Ayano have three arrays of integers $a_1, \ldots, a_n$, $b_1, \ldots, b_n$ and $c_1, \ldots, c_n$. Initially, the value of every b_i, c_i is zero.

Now she wants you to do q operations. There are two types of operations:

- `1 l r w` $(1 \le l \le r \le n,\ 1 \le w \le n)$: for each i that $l \le i \le r$, set a_i to w.
- `2 l r w` $(1 \le l \le r \le n,\ 1 \le w \le 10^9)$: for each i that $l \le i \le r$, increase c_i by w.

At the end of each operation, for each i $(1 \le i \le n)$, Ayano will increase b_{a_i} by c_i.

Please tell her the array $b_1, \ldots, b_n$ after all of the operations. Because the answer is very large, you only have to output each number modulo 2^{64}.

Input

The first line contains two integers n and q $(1 \le n, q \le 5 \cdot 10^5)$, representing the length of the arrays and the number of operations.

The next line contains n integers $a_1, \ldots, a_n$ $(1 \le a_i \le n)$.

Each line of the following q lines contains four integers t_i, l_i, r_i, w_i $(1 \le t_i \le 2)$ — the operations.

Output

Output one line containing n integers. the i-th integer represents b_i modulo 2^{64}.

Example

standard input	standard output

5 6
1 2 3 4 5
2 2 4 1
1 2 3 3
2 3 4 3
1 3 5 4
2 1 5 2
1 1 3 2
	2 12 12 36 0

Problem C. Customs Controls 2

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

Look how short the statement is! This must be the easiest problem.

Given a directed acyclic graph G, you need to assign each vertex i a positive integer weight w_i. Your goal is to make all paths from 1 to n of equal length.

A directed acyclic graph is a graph with directed edges and without cycles.

The length of a path is defined as the sum of the weights of vertices on the path.

Input

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

For each testcase:

- The first line contains two integers n, m ($1 \le n \le 2 \cdot 10^5$, $1 \le m \le 5 \cdot 10^5$), denoting the number of vertices and edges.
- The next m lines each contains two integers u, v, denoting an edge from u to v.

It is guaranteed that $\sum n \le 2 \cdot 10^5$, $\sum m \le 5 \cdot 10^5$.

It is guaranteed that the graph contains no multiple edges, no self-loops and no cycles. It is also guaranteed that every vertex is reachable from 1 and can reach n.

Output

For each testcase, if there is no solution, then output "`No`" on a single line. Otherwise, output "`Yes`" on the first line, then n positive integers $w_1, w_2, \ldots, w_n$ ($1 \le w_i \le 10^9$) on the second line.

Examples

standard input
```
2
3 3
1 2
1 3
2 3
8 9
1 2
1 3
1 4
2 5
3 6
4 7
5 8
6 8
7 8
```

standard output
```
No
Yes
1 1 2 3 3 2 1 1
```

standard input
```
2
11 16
1 2
1 3
1 4
1 5
2 6
4 6
3 7
4 7
5 8
6 8
2 9
3 9
7 10
8 10
9 11
10 11
8 10
1 2
1 3
2 4
3 5
3 6
4 6
2 7
5 7
6 8
7 8
```

standard output
```
Yes
1 1 1 1 2 1 2 1 3 1 1
No
```

Problem D. Digits

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

Richard loves range sums. Anita loves palindromes.

Richard has an array of n non-negative integers $a_1, a_2, \ldots, a_n$. He wants to split it into several segments and then calculate the range sums of each segment.

Anita wonders the number of partitions that the **digits** of the range sums form a palindrome. A string is a palindrome, if and only if it is the same as its reverse.

For example, array $a = [1, 1, 4, 5, 1, 4, 1, 9, 1, 9]$ can be partitioned into 3 segments $[1, 1, 4, 5], [1, 4, 1, 9, 1], [9]$ and the sum of these segments are $11, 16, 9$, therefore this partition results in a string 11169.

Note that the sum always doesn't contain prefix zeroes. That means the same partition will result in only one possible string.

Richard's array is too large, so Anita can't calculate the answer and cried. Can you help her?

Since the answer can be too large, you only need to answer it modulo $10^9 + 9$.

Input

Each test contains multiple test cases.

The first line contains one single integer t ($1 \le t \le 100$). The description of the t test cases follows.

The first line of each test case contains an integer n ($1 \le n \le 150$), the size of array a.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 666666$).

It is guaranteed that the sum of n of all test cases doesn't exceed 200.

Output

For each test case, output an integer in one line, representing your answer modulo $10^9 + 9$.

Example

standard input	standard output

5
2
123456 654321
5
0 0 0 0 0
6
1 1 1 1 1 1
7
1 1 4 5 1 4 1
9
1 2 3 1 1 1 2 1 1
	2
	16
	8
	4
	6

Note

In the first case, partition [123456], [654321] results in 123456654321, and partition [123456, 654321] results in 777777, each of which is a palindrome.

In the second case, any partition will result in a string with only zeroes, obviously a palindrome. so the answer is $2^4 = 16$.

In the third case, valid partitions are

$[1], [1], [1], [1], [1], [1] \to 111111,$

$[1], [1], [1, 1], [1], [1] \to 11211,$

$[1], [1, 1], [1, 1], [1] \to 1221,$

$[1], [1, 1, 1, 1], [1] \to 141,$

$[1, 1], [1], [1], [1, 1] \to 2112,$

$[1, 1], [1, 1], [1, 1] \to 222,$

$[1, 1, 1], [1, 1, 1] \to 33,$

$[1, 1, 1, 1, 1, 1] \to 6.$

Problem E. Elevator

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

There are n elevators participating in a speed race starting at second 0 in a building of m floors numbered 1 through m.

The i-th elevator will start at second x_i at floor 1 and will ascend at the speed of 1 floor per second. Besides, there is a button on each floor **except floor** 1 **and floor** m, which, if **pressed**, will make the first elevator that reaches this floor stop for 1 second. If more than one elevator reaches a floor at the same time, only the elevator with the smallest index will be regarded as the first one. There are **no** buttons pressed now, but you can press the buttons on some floors before the start of the race. Notice that you can **not** press the button after the race starts.

Now you wonder whether you will be able to manipulate the race by pressing the buttons to make the i-th elevator be the first to reach floor m, and how many buttons you have to press at least if you can. If more than one elevator reaches floor m at the same time, only the elevator with the smallest index will be regarded as the first one.

Input

The first line contains two positive integers n, m ($1 \le n \le 5 \cdot 10^5, 2 \le m \le 10^9$), denoting the number of elevators and floors.

The second line contains n positive integers $x_1, \ldots, x_n$ ($1 \le x_i \le 10^9$), denoting the start time of the elevators.

Output

Output n lines. The i-th line should contain the minimum number of buttons you have to press to make the i-th elevator to be the first to reach floor m. Output -1 instead if it is impossible.

Example

standard input	standard output
6 20 3 8 12 6 9 9	0 8 -1 4 13 14

Problem F. Equations

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2.5 seconds
Memory limit:	256 megabytes

Let $f(a, b, m)$ be the **least non-negative** solution of the congruence equation:

$$ax \equiv b \pmod m$$

If there isn't a solution, then $f(a, b, m) = 0$.

Given n, a, b, calculate $\sum_{i=1}^{n} f(a, b, i) \bmod 998244353$.

Input

The first line contains an integer T $(1 \le T \le 5)$ denoting the number of test cases.

Then T lines follow, each containing three positive integers n, a, b $(1 \le n \le 10^{18}, 1 \le a, b \le 10^6)$.

Output

Output T lines. The i-th line contains the answer of the i-th test case.

Example

standard input	standard output

5
5 4 3
5 3 4
10 5 8
10 8 5
100 79 97
	2
	3
	15
	10
	2519

Problem G. Game

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

Once upon a time, there were two saints named St. Alice and St. Bob.

Being saints were quite boring, so they decided to play a game. The game was about divisibility.

There were two sets of integers, called A and B. And there were two integers α, β. In the beginning, $\alpha = \beta = 1$.

Then, they take turns. **First of all, Alice takes a turn**, then **Alice** goes first, then Bob, then Alice, then Bob, then Alice, and so on.

In Alice's turn, she must choose a number x from A, and change α to $\alpha \times x$.

In Bob's turn, he must choose a number x from B, and change β to $\beta \times x$.

If at some time $\alpha \mid \beta$ (which means there exists an integer k such that $k \times \alpha = \beta$), Bob wins. Alice cannot win, she only wanted to play the game forever.

The saints were smart, so both players made optimal moves (to win or to not lose).

After some experiments, Alice found the game too easy, so she wanted to delete a subset of A while keeping herself unbeatable. Call a subset of A *valid*, if and only if deleting it does not affect Alice's unbeatable position.

Saints live long, and they kept playing. Alice asks you to find the number of valid subsets of A, could you do it?

Input

The first line contains two integers $|A|, |B| (1 \le |A|, |B| \le 500)$ — numbers of elements of A, B, respectively.

The second line cantains $|A|$ integers a_i $(2 \le a_i \le 500, a_i < a_{i+1})$ — the elements of A.

The third line cantains $|B|$ integers b_i $(2 \le b_i \le 500, b_i < b_{i+1})$ — the elements of B.

Output

Output a single integer — the number of valid subsets of A, modulo $10^9 + 7$.

Bob may win even if Alice deletes nothing, which means the saints are making fun of you, but print 0 anyway.

Examples

standard input	standard output

`2 3`
`2 6`
`6 7 8`
	`2`

`6 13`
`6 7 12 18 21 29`
`8 11 12 13 15 16 20 21 23 24 27 28 30`
	`56`

Note

In the first example, $\{2\}, \{\}$ are the only two valid subsets.

Problem H. GameX

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

Once upon a time, there were two saints named St. Alice and St. Bob.

Being saints were quite boring, so they decided to play a game. The game was about the MEX operation, and was therefore named GameX.

To help you, a mere mortal, to understand the game, we first present the definition of MEX. Given a set S of integers, define $\mathrm{MEX}(S)$ as the smallest natural number which is not in S. In other words, $\mathrm{MEX}(S) = \min\{x \in \mathbb{N} \mid x \notin S\}$.

The game went as follows.

Before the game started, $S = \{a_1, a_2, \cdots, a_n\}$, which contained the Secret of Life, the Universe and Everything.

The two saints moved alternately, with St. Alice being the first. During one's move, he/she could choose an arbitrary integer x, and insert x into S. So S is updated to $S \cup \{x\}$.

After k rounds, each player made k updates, and now it's time to decide the winner. St. Alice wins iff $\mathrm{MEX}(S)$ is even, and Bob wins otherwise.

Saints are very smart, so both of them made optimal moves. Can a mortal like you decide the winner?

Input

The first line contains a positive integer T ($1 \le T \le 10^4$), denoting the number of testcases.

For each testcase:

- The first line contains two integers n, k ($1 \le n, k \le 2 \times 10^5$), denoting the size of S before the game started and the number of rounds.
- The next line contains n distinct natural numbers $a_1, a_2, \cdots, a_n$ ($0 \le a_i \le 10^6$), denoting S.

It is guaranteed that $\sum n, \sum k \le 2 \times 10^5$.

Output

For each testcase, output one line consisting of the name of the winner. If St. Alice won output `Alice`, otherwise output `Bob`.

Example

standard input

```
5
14 5
7 13 1 6 14 2 16 17 18 19 34 36 20 23
13 5
8 10 3 13 14 15 16 17 18 19 20 36 38
14 5
14 20 12 6 0 16 8 11 9 17 13 3 5 19
14 5
15 7 13 3 1 17 16 14 0 12 4 10 22 53
14 5
7 3 4 0 14 15 16 17 18 19 20 21 22 23
```

standard output

```
Bob
Bob
Alice
Bob
Alice
```

Problem I. Infection

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

A highly propagating bacterium infects a tree of n nodes (with $n-1$ edges, no cycles). These nodes are indexed from 1 to n.

Exactly one node will be infected at the beginning. Each node on the tree has an initial susceptibility weight a_i, which represents that node i has a probability of $\frac{a_i}{\sum_{i=1}^{n} a_i}$ to become the initial infected node of the tree.

In addition, each node has an infection probability p_i, which represents its probability of being infected by adjacent nodes.

Specifically, starting from the initial infected node, if a node is infected, the uninfected node x that is adjacent to it will have a probability of p_x to become a new infected node, and then x can continue to infect its adjacent nodes.

Now, your task is to calculate the probability that exactly k nodes are eventually infected. You need to output an answer for each $k = 1, \ldots, n$.

You need to output the answer modulo 10^9+7, which means if your answer is $\frac{a}{b}$ ($\gcd(a,b)=1$), you need to output $a \cdot b^{-1} \bmod 10^9+7$, where $b \cdot b^{-1} \equiv 1 \pmod{10^9+7}$.

Input

The first line contains an integer n ($2 \le n \le 2\,000$), denoting the number of nodes of the tree.

The next $n-1$ lines, each line contains two positive integers u and v ($1 \le u, v \le n$), denoting that there is an edge (u, v) on the tree.

Next n lines, the i-th line contains three positive integers a_i, b_i, c_i, where a_i means as above and $p_i = \frac{b_i}{c_i}$.

It is guaranteed that $1 \le a_i, b_i, c_i \le 10^9, \sum_{i=1}^{n} a_i \le 10^9, b_i \le c_i, \gcd(b_i, c_i) = 1$.

Output

Output n lines, each line contains single integer. The integer on the i-th line should be the answer modulo 10^9+7 when $k = i$.

Example

standard input	standard output

5
2 1
5 2
3 2
4 3
2 1 5
3 1 2
2 1 1
2 1 1
3 1 2
	208333335
	166666668
	166666668
	950000007
	508333337

Problem J. Math Exam

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

There are many integer sequences in the world, and your mission is to find how many sequences are good.

An integer sequence a_i of length n is good if and only if all of these conditions holds:

- $\forall i \in [1, n], 4S_i = {a_i}^2 + 2a_i + 1.$
- $\forall i \in [1, n], |a_i| \le m.$

Where $S_i = \sum_{j=1}^{i} a_j$.

You will be given n and m, and it is guaranteed that m is **odd**.

Since the answer may be very large, you should calculate it modulo $998\,244\,353$.

Input

The only line contains two integers n and m ($1 \le n \le 10^7$, $1 \le m \le 2n$, m is odd).

Output

Output a single integer — the number of good sequences meeting the constraints, modulo $998\,244\,353$.

Examples

standard input	standard output
`9 13`	`124`
`500 999`	`195157058`

Problem K. Middle Point Graph

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

You're given a simple connected undirected graph with n vertices and m edges.

For each vertex, we assign it a random point (x_i, y_i, z_i), where x_i, y_i, z_i are independent uniform random real numbers in $[0, 1]$.

For each edge, its coordinate is defined as the middle point of its two ends' coordinates. The middle point of (a, b, c) and (x, y, z) is $(\frac{a+x}{2}, \frac{b+y}{2}, \frac{c+z}{2})$.

Among these $n + m$ points, you are to find the expected number of ways to choose 4 coplanar distinct points. Print the answer modulo $10^9 + 7$.

Input

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

For each testcase:

- The first line contains two integers n, m, $(1 \le n \le 2 \cdot 10^5, n - 1 \le m \le 5 \cdot 10^5)$ denoting the number of vertices and edges.
- The next m lines each contains two integers u, v $(1 \le u, v \le n)$, denoting an edge connecting u and v.

It is guaranteed that $\sum n \le 2 \cdot 10^5$, $\sum m \le 5 \cdot 10^5$.

An empty line is placed before each testcase for better readability.

Output

For each testcase, output one line containing a single integer denoting the answer module $10^9 + 7$.

Example

standard input	standard output

3
 	
7 18
2 1
2 3
3 4
2 5
6 4
7 5
6 5
3 1
1 5
1 7
7 3
2 6
2 7
4 5
5 3
4 2
6 7
6 3
 	
5 7
1 2
2 3
4 2
5 1
1 4
3 5
3 1
 	
1 0
	593
	88
	0

Problem L. Station of Fate

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

There are n people standing in m stations, forming m queues.

You don't know which person is in which station, or in what order they are standing in queue, so you want to count the number of different **schemes**. Two schemes are considered different, if and only if there exists a station whose queue consists of different people, or has different orders.

Calculate the number of different schemes modulo 998 244 353.

Input

The first line contains an integer T $(1 \le T \le 100)$, denoting the number of test cases. Then T test cases follow.

Each test case contains a single line containing two integers n, m $(1 \le m \le n \le 10^5)$.

Output

For each test case, output the number of different schemes modulo 998 244 353.

Example

standard input	standard output

2
3 2
6 3
	12
	7200

Problem M. XOR Sum

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

Louis loves integers. He defines the value of a sequence of non-negative integers $A = [a_1, a_2, \ldots, a_k]$ as the following formula ($\oplus$ means bitwise exclusive-or):

$$\sum_{i=1}^{k} \sum_{j=1}^{i-1} a_i \oplus a_j$$

He wants to know how many different sequences A holds the following conditions:

- The length of A is k.
- The value of A is n.
- $0 \le a_i \le m$ $(1 \le i \le k)$.

Two sequences $[a_1, \ldots, a_k]$, $[b_1, \ldots, b_k]$ are considered different if and only if there exists an index i such that $a_i \ne b_i$. Tell Louis the answer module $10^9 + 7$.

Input

The input contains of single line containing three integers n, m, k, $(0 \le n \le 10^{15}$, $0 \le m \le 10^{12}$, $1 \le k \le 18)$.

Output

Output the answer modulo $10^9 + 7$.

Examples

standard input	standard output
`6 2 3`	`12`
`30 6 5`	`1520`

Note

In the first example, vaild sequences are

$[0, 1, 2], [0, 2, 1], [1, 0, 2], [2, 0, 1], [1, 2, 0], [2, 1, 0], [1, 1, 2], [1, 2, 1], [2, 1, 1], [2, 2, 1], [2, 1, 2], [1, 2, 2]$.