

Problem A. Rajae in the Kitchen

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

Today is Rajae's birthday!! *Who Asked?*

Sadly, no one remembered his birthday, so he decided to make a birthday cake by himself. He went searching in the kitchen for suitable ingredients, but unfortunately, he found nothing but an array of positive integer numbers. He is going to make a cake out of this array using the following technique:

- Firstly, he divides the array into several consecutive non-empty subarrays.
- Every element of the array should belong to exactly one subarray.
- Then for each subarray, he creates an edge of length equal to the sum of the numbers in the subarray.
- Then he forms a cake in the shape of a convex polygon using **all the edges he created**.

Rajae is wondering in how many ways he can divide the array into smaller subarrays such that he can make a cake out of it. Rajae is too bad at maths so he needs your help to know the number of ways modulo $10^9 + 7$.

Two ways are considered different if there exists an element with index i belonging to two different subarrays in each way.

Input

The first line contains a single integer number N ($3 \leq N \leq 10^6$), the size of the array.

The second line contains N integer numbers $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 10^9$), the elements of the array.

Output

Output the number of ways modulo $10^9 + 7$.

Examples

standard input	standard output
3 3 4 5	1
6 1 5 2 4 1 2	18
10 7 2 3 5 8 3 1 9 4 5	436

Problem B. Omar's Magic Trick

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

Omar and Ahmed are playing with an infinite deck of cards. Each card has a single-digit number written on it (ranging from 1 to 9).

Ahmed teaches Omar a new trick. Here are the steps:

1. **Initial Selection:**

- Omar selects three cards from the deck.

2. **Transformation:**

- Ahmed instructs Omar to multiply the value of each selected card by 3.
- Omar then replaces each card with the digits of the result. For example, if the original card value is 8, Omar replaces it with two cards: 2 and 4.
- Omar repeats this transformation a total of n times.

3. **Final Choice:**

- After all the transformations, Omar chooses a card from the last set of cards (the result of the n_{th} transformation) and hides it.

Given the remaining cards in the deck (after all transformations) except the hidden card, can you determine the hidden one?

It is guaranteed that the answer always exists

The order of the number is not important, for example the [1 8 1 5 6 2 1] and [1 8 1 5 6 1 2] give the same answer which is 3

Input

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

The first line of each test case contains 2 integers n, m ($2 \leq n \leq 33, 2 \leq m \leq 10^6$).

The second line contains m integers $a_1, a_2, \dots, a_m$ ($1 \leq a_i \leq 9$).

An additional constraint on the input: the sum of m over all test cases doesn't exceed 10^6 .

It is guaranteed that the set of cards is always a valid set

Output

For each test case, print a single integer — The card Omar has hidden.

Example

standard input	standard output
1 2 7 1 8 1 5 6 2 1	3

Note

In the **first** test case:

- The initial set is 2, 5, 9.
- After the first operation, the set becomes 6, 1, 5, 2, 7.
- After the second operation, the set becomes 1, 8, 3, 1, 5, 6, 2, 1.
- Omar hides 3.
- So the remaining set is 1, 8, 1, 5, 6, 2, 1.

Problem C. Hungry Horse

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

Yaman, the proud owner of a spirited Arabian horse, set out on a journey through the enchanting desert. Aware of his horse's hearty appetite, Yaman scattered food dishes along the path, each with its own eating time and position.

The food dishes are numbered from 1 to n , and each food dish has two numbers p_i and t_i , the position and the number of minutes needed to eat the i -th dish respectively.

The horse starts at position 0 and it can move forward and can't go back. Also, every step the horse takes forward would consume an additional minute.

As is known, the Arabian horse is intelligent, so it will choose the maximum number of dishes possible to eat during the D -minute journey.

With time ticking away, Yaman aimed to determine the maximum number of dishes his Arabian horse could relish during the journey. Help Yaman determine it.

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^5$). A description of the test cases follows.
The first line of each test case contains three integers n ($1 \leq n \leq 10^5$), ($1 \leq D \leq 10^9$) — the number of food dishes and the journey duration in minutes respectively.
The next line contains n integers p_i ($1 \leq p_i \leq 10^9$) — the position of the i -th dish.
The next line contains n integers t_i ($1 \leq t_i \leq 10^9$) — the number of minutes needed to eat the i -th dish.
It is guaranteed that the sum of n over all test cases does not exceed 10^5 , and it is guaranteed that all positions p_i are **unique**.

Output

For each test case, print the maximum number of dishes that the horse could eat.

Example

standard input	standard output
2	4
5 10	5
1 2 3 4 5	
1 1 1 1 2	
5 10	
1 2 3 4 5	
1 1 1 1 1	

Problem D. Switching To Windows

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

Uncle Eyad is a Linux creep and his friend Jaber has always begged him to switch to Windows and he finally did!! Eyad works for the uncleship company owned and managed by his boss Hosen the grand uncle, and he is in control of the company's database system. Hosen usually gives Eyad queries to the database, and Eyad usually responds to them and maintains the database easily using Linux, but not anymore!!

As a good friend of Eyad, he asked you to help him respond to Hosen's queries and maintain the database accordingly.

Initially, the database is empty, Hosen gives Eyad Q unclly queries one by one, and the queries have 5 different types as described in the following:

- 1 $S V$: Add a string S whose value is V to the database.
- 2 i : Erase the string added in the i -th query from the database. It is guaranteed that the i -th query was of the first type and that the string is currently in the database.
- 3 $L R x$: For each string in the database that was added in the i -th query ($L \leq i \leq R$), change its value to x .
- 4 $L R x$: For each string in the database that was added in the i -th query ($L \leq i \leq R$), increase its value by an amount of x .
- 5 $S L R$: Output the number of substrings in S which are currently in the database and have a value V ($L \leq V \leq R$). The substrings are not necessarily distinct.

Input

The first line contains a single integer number Q ($1 \leq Q \leq 2 * 10^5$). The following Q lines contain the description of the queries in the previous format in the problem statement.

- 1 $S V$: ($1 \leq V \leq 10^{18}$).
- 2 i : ($1 \leq i \leq 2 * 10^5$).
- 3 $L R x$: ($1 \leq L, R \leq 2 * 10^5$), ($1 \leq x \leq 10^{18}$).
- 4 $L R x$: ($1 \leq L, R \leq 2 * 10^5$), ($1 \leq x \leq 10^9$).
- 5 $S L R$: ($1 \leq L, R \leq 10^{18}$).

It is guaranteed that the sum of lengths of all strings in the input does not exceed $2 * 10^5$, and all the strings given in the input in the first type query are **distinct**, and it is guaranteed that the first query is a first-type query.

Output

For each query of the 5-th type, output the answer to it.

Example

standard input	standard output
13	4
1 a 3	3
1 b 6	0
1 ab 5	1
1 ca 2	
1 ac 10	
1 aba 1	
5 caba 3 6	
2 2	
5 caba 3 6	
3 1 4 2	
5 caba 3 6	
4 6 6 5	
5 caba 3 6	

Problem E. Zero Hour

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

George enjoys playing old classic games. One day, he started playing the popular game called “General Zero Hour” on his computer. While playing the game, he encountered a problem and had to stop.

In the game, George has a certain number of soldiers, denoted by n . Each soldier is assigned a power level, denoted by a_i . Now, George needs to divide these soldiers into groups, while adhering to the following two conditions:

- Each soldier should belong to exactly one group.
- If two soldiers x and y belong to the same group, then the Bitwise XOR value of their power levels should be greater than or equal to the minimum power level between them, In other words $(a_x \oplus a_y)$ should be greater than or equal to $\min(a_x, a_y)$.

George wants to minimize the number of groups required to satisfy these conditions because he is too lazy to name all the groups.

Can you help George with this problem so he can continue playing?

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^5$). A description of the test cases follows.
The first line of each test case contains a single integer n ($1 \leq n \leq 10^5$) indicating the number of soldiers.
The second line of each test case contains n integers $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 2^{60}$) , a_i indicates the power of the soldier i .
It is guaranteed that the sum of n over all test cases does not exceed 10^5 .

Output

For each test case, print the minimum number of groups.

Example

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

Problem F. Legend Whispers

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

In the old city of Damascus, two ancient arrays, a and b , each of length n , have been discovered by the greatest professor in his prime, George The Great

The legend whispers of a unique measure of beauty for these arrays:

- If we cannot shuffle the array a so that a_i is divisible by b_i for every $1 \leq i \leq n$, the beauty is messed up and becomes -1
- However, If we can shuffle the array a so that a_i is divisible by b_i for every $1 \leq i \leq n$, the beauty of the arrays will be equal to the

$$\min_{1 \leq i \leq n} \left\{ \frac{a_i}{b_i} \right\}$$

We ask you to print the maximal beauty you can get of the two arrays after rearrangement of the array a .

Input

The first line contains one integer number ($1 \leq T \leq 100$) the number of test cases.

The first line of each test case contains an integer ($1 \leq n \leq 500$).

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^6$). The third line contains n integers $b_1, b_2, \dots, b_n$ ($1 \leq b_i \leq 10^6$).

An additional constraint on the input: the sum of n over all test cases does not exceed 500.

Output

For each test case, print a single integer — The maximal beauty of a and b , or -1 if there is no suitable arrangement.

Example

standard input	standard output
2	4
3	-1
6 10 12	
1 2 3	
3	
1 2 3	
3 3 3	

Problem G. Intersection Not Allowed

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

In the city of contests, Yaman managed a popular storage facility called “Secure Storages“ with a range of lockers numbered from 1 to n . Contestants relied on Secure Storages to securely store their belongings.

One day, Yaman received q requests from individuals, each with an ID from 1 to m . These requests specified the desired locker number and a time interval for rental.

In other words, Yaman received q requests, and each request contains three numbers x, l, r , which means that the locker x will be rented at the time $[l, r]$.

Yaman faced the challenge of assigning lockers without any intersections or conflicts and was determined to provide excellent service. Yaman carefully reviewed the requests. He aimed to delete **at most** one request, allowing for a seamless locker assignment process. Yaman’s goal was to ensure that no locker was shared by multiple requests at the same time.

Help Yaman determine if he can achieve his goal.

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^3$). A description of the test cases follows.
The first line of each test case contains two integers n, q ($1 \leq n, q \leq 10^3$), — the number of lockers, and the number of requests, respectively.
Each of the next q lines contains three integers x_i, l_i, r_i ($1 \leq x_i \leq n$), ($1 \leq l \leq r \leq 10^3$) — the representation of the i -th request.
It is guaranteed that the sum of q and the sum of n over all test cases does not exceed 10^3 .

Output

For each test case, print “YES“ (without quotes) if Yaman can achieve his goal; otherwise, you have to print “NO“ (without quotes).

Example

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

Problem H. One Punch MEX

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

The contest arena is under attack ! George wants to save it, so he will use his special ability called “One Punch MEX”.

George has n power stones numbered from 1 to n . For n times the following action will happen in order:

1. Uniformly at random, one of the stones will disappear
2. The MEX of the remaining stones will add to the One Punch MEX damage

George is wondering what will be the expected damage of the One Punch MEX if he has n power stones
MEX is the minimum **positive** number not present in the set. MEX of $[]$ is 1, MEX of $[3, 5, 1]$ is 2

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^5$). A description of the test cases follows
The first line of each test case contains a single integer n ($1 \leq n \leq 10^5$) indicating the number of power stones
It is guaranteed that the sum of n over all test cases does not exceed 10^5

Output

For each test case, output a single line containing the expected damage of One Punch MEX modulo 998244353

Example

standard input	standard output
2	1
1	499122179
2	

Note

In the second test case:
George has stones $[1, 2]$
With the probability $\frac{1}{2}$, the first stone will be removed, then the second stone with the sum of MEX equal to 2
With the probability $\frac{1}{2}$, the second stone will be removed, then the first stone with the sum of MEX equal to 3
The expected damage of One Punch MEX is $\frac{5}{2}$

Problem I. Nested Circles

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 1024 megabytes

Since coach *Attal* loves geometry problems, he tried to make an interesting problem to discuss it with DCPC community coaches. After discussing it, they came up with a simple solution. So they will tell you the problem.

You are given n circles numbered from 1 to n . Each circle is defined by an integer center (x_i, y_i) and an integer radius r_i .

Then we will ask you q questions. In each question, we will give you an integer point (x_i, y_i) , and you have to find the number of circles that cover this point.

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^5$). A description of the test cases follows.
The first line of each test case contains two integers n, q ($1 \leq n, q \leq 10^5$) — the number of circles, the number of questions, respectively.
Each of the next n lines contains three integers x_i, y_i, r_i ($1 \leq x_i, y_i \leq 10^5$), ($1 \leq r_i \leq 10$) — the center and the radius of the i -th circle.
Each of the next q lines contains two integers x_i, y_i ($1 \leq x_i, y_i \leq 10^5$) — the point of the i -th question.
It is guaranteed that the sum of q and n over all test cases does not exceed 10^5 .

Output

For each question i , print the number of circles that cover the point in the i -th question.

Example

standard input	standard output
2	1
3 3	1
1 1 1	1
2 2 1	1
3 3 1	0
1 1	0
2 2	1
3 3	2
3 5	
1 2 3	
2 3 2	
3 4 1	
1 1	
4 5	
8 10	
4 3	
3 2	

Problem J. Epic Fight

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

George is so excited to see an epic fight between the legends *Ahmad* and *Yaman*.

George gives them n cells that are numbered from 1 to n where each cell i and $i + 1$ ($1 \leq i \leq n - 1$) are connected by a two-way road to play a game on them.

The game is going as follows: First of all, *Yaman* will choose a cell to put an egg on it and mark it as a visited cell, then they will take turns starting from *Ahmad* as follows:

- The player will move the egg to an adjacent cell that is not visited yet and mark it as visited.

The player who can't move the egg will **lose** the game.

Now *Yaman* is wondering how many cells he can choose so he can win.

Input

The first line contains the number of test cases t ($1 \leq t \leq 10^5$). A description of the test cases follows.

The first line of each test case contains a single integer n ($1 \leq n \leq 10^{18}$) indicating the number of cells.

Output

A single integer represents the number of winning starting cells for *Yaman*.

Example

standard input	standard output
4	1
1	2
3	389
777	0
2	

Problem K. Marks

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

The final exams season of school is so close, and Ahmad is a teacher in some school, and his task is extremely boring. He has to sum up the marks of every student in all subjects to find the total score of the students. He wants help from you in summing up the marks. He will provide you the marks of n students, and you have to sum them up for him. There are 8 subjects in the school, so for every student, you will be given 8 integers, and you have to print the total score.

Input

The first line contains an integer t ($1 \leq t \leq 100$), the number of test cases.

Every test case consists of $n + 1$ lines. The first contains an integer n ($1 \leq n \leq 100$), the number of students that Ahmad will provide their marks.

Each of the next n lines contains 8 integers $[mark_{i1}, mark_{i2}, \dots, mark_{i8}]$ ($1 \leq mark_{ij} \leq 100$) — the mark of the i -th student in the j -th subject.

Output

For every test case, you have to print n integers each on a line, the total score of the student.

Example

standard input	standard output
1	36
2	45
1 2 3 4 5 6 7 8	
10 3 4 5 6 7 8 2	

Problem L. BBS Queries

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 1024 megabytes

You are given a balanced bracket sequence of only '(' and ')' brackets of length $2n$, a value of a_i is attached to every bracket, and for every two matched pair of brackets, they have the same value

For example, let the balanced bracket sequence be: $()(())()$ and the array a : $[1, 1, 5, 1, 1, 5, 2, 2]$, Notice that values at indexes 1 and 2 are same because the bracket of index 2 is the one that closes the bracket at index 1, values at indexes 3 and 6 are same because the bracket of the index 6 is the one the closes the bracket at index 3, and so on for every two matched pair in the balanced bracket sequence

You are also given two types of queries:

The query of type 1 is as follows: $1\ l_1\ r_1\ l_2\ r_2\ v$, where l_1 and r_1 are indexes of two matched brackets, l_2 and r_2 are also indexes of two matched brackets, in such query you have to add a value of v to the value of every two matched brackets which their opening bracket is before or equal to $\min(l_1, l_2)$ and closing bracket is after or equal to $\max(r_1, r_2)$

Please note that it isn't necessary that $l_1 \leq l_2$ nor $r_1 \leq r_2$

The query of type 2 is as follows: $2\ l\ r$, where l and r are indexes of two matched brackets, in such query you have to find the value of these two brackets, (note that always any two matched brackets will have the same value and you are asked to find this value, not the sum of values of the two brackets)

Refer to the notes and check the described example for better understanding

A **balanced bracket sequence** is a string consisting of only brackets, such that this sequence when inserted with certain numbers and mathematical operations, gives a valid mathematical expression

Input

The first contains an integer t ($1 \leq t \leq 10^5$) the number of testcases

In every test case you will read two integers ($1 \leq n \leq 5 * 10^5$) and ($1 \leq q \leq 5 * 10^5$), where $2n$ is the length of the string and q is the number of queries

The second line of every testcase contains the string of brackets balanced bracket sequence, the length of the string is $2n$,

The third line of every testcase contains the array a ($0 \leq a_i \leq 10^9$), the values attached to the brackets

The following q lines contains the q queries

It's guaranteed that all l s and r s in queries are between 1 and $2n$, and the string forms a balanced bracket sequence with brackets of type $()$ only (doesn't contains brackets of other types like $[]$ or $\{ \}$)

It's guaranteed that the sum of n and q all test cases don't exceed $5 * 10^5$

Output

For every testcase, for each query of type 2 you have to print its answer as described above

Example

standard input	standard output
1	5
5 4	8
((()())())	1
3 1 1 5 1 1 5 2 2 3	
2 4 7	
1 2 3 4 7 5	
2 1 10	
2 2 3	

Note

Description of the given sample:

First query: 4 7

The value of the matched pair at indexes 4 and 7 is 5

Second query: 2 3 4 7 5

The matched pair at indexes 1 10 is the only matched pair of brackets that includes pairs 2 3 and 4 7 together, hence we increase its value by 5 and the array of values after this query becomes: 8 1 1 5 1 1 5 2 2 8

Third query: 1 10

The value of the matched pair at indexes 1 10 is 8

Forth query: 2 3

The value of the matched pair at indexes 2 3 is 1

Problem M. ACPC

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

Both Arab Collegiate Programming Contest and Aleppo Collegiate Programming Contest can be abbreviated with ACPC which could lead to ambiguity, but when you are in Aleppo Collegiate Programming Contest then ACPC should mean Aleppo Collegiate Programming Contest, one of your friends doesn't know that so you have to tell him what is the correct meaning of ACPC, with knowing that you are at Aleppo Collegiate Programming Contest

Input

The input contains one string s ($1 \leq |s| \leq 10$), the name of your friend

Output

The output should be "Aleppo"(without quotes)

Examples

standard input	standard output
ahmad	Aleppo
hossain	Aleppo