Problem A

3 Friends

You and your 2 other friends want to participate in a program. You all live in different cities in the country. Cities are connected by roads. Every city is reachable from every other city by exactly one path. **The program venue is in city 1.** You can take a bus to go from one city to another adjacent city. More formally between every two adjacent cities, a bus route is available. To get into the bus you need to buy tickets. So if you want to travel from city p to city q and distance between city p and q is r then you have to buy r tickets to complete the trip. You can travel in a group. The cost of tickets is a bit different in your country. You have to pay **X** dollars for a group of **1** member, **Y** dollars for a group of **2** members and **Z** dollars for a group of **3** members. You and your friends are students. So you want to minimize the cost. To minimize the cost you or your friend may meet in another city to make a group and go to city 1 altogether. If you want to make a group, you can only add members to the group, but you cannot remove members from the group. Suppose you and one of your friends may create a 2 members group in some city and go to City 1 together or go to another city, meet with your other friend make a new group of 3 members, then go to city 1. Anyone can create a group in any city.

Input

The first line contains an integer **T** denoting the number of test cases.

In the first line of each test case, there will be two integers N ($3 \le N \le 10^5$) and Q ($1 \le Q \le 10^5$), denoting the number of cities in the country and the number of queries.

Then there will be 3 integers X, Y, Z ($0 \le X, Y, Z \le 10^9$) in one line.

Then there will be **N-1** lines containing 2 integers u, v, ($1 \le u,v \le N$) means city **u** and city **v** is connected.

The next Q line will have 3 integers **a, b, c**. **a** is your city, **b** and **c** are your friend's city.

The sum of all **N** in a file is $\le 3*10^5$

The sum of all **Q** in a file is $\le 2*10^5$

Output

For each test case print case number in one line.

For each query, you have to output the minimum cost to reach city 1 including you and your 2 other friends.

Sample Input

```
1
7 3
3 5 7
1 2
1 3
2 4
2 5
3 6
3 7
2 5 7
4 5 7
3 6 6
```

Output for Sample Input

```
Case 1:
14
17
12
```

Problem B

Food Blogging

Rakib wants to open a food blog. He believes that he has a great sense of taste and the world would be lucky to have his food reviews. He has spent the last few months going to different restaurants and has created a massive list of restaurants with their ratings. When he showed it to his friend Apon, Apon said that just a rating with just a number looks very boring. He suggested that if he broke down his rating into various categories such as Ambience, Service, Taste, etc, his blog would be a lot more interesting. But Rakib has already spent a lot of time making the list. He doesn't want to go through the trouble of making it all over again. So he comes up with a plan.

For each rating, he will break it into different categories. He wants to make sure that after breaking the ratings into categories, all the orders of the ratings are preserved, i.e. if a restaurant is ranked higher than another before breaking its rating into categories, it should also be higher after breaking it. Similarly, if two restaurants have the same rating before being down into categories, they should have the same rank after the breakdown. Now he realizes that the importance of each category is not the same, so he assigns weights to each category. He wants to make sure that he doesn't get caught being lazy, so he wants to make sure that there are no two restaurants such that they have the exact same points as one other in each and every category. The rank of a restaurant is the weighted sum of the points it has in each category.

With this new scheme, Rakib is ready to begin his work. But breaking down so many ratings will be very time consuming, so he has come to you for help. Can you help him break his ratings into categories, keeping the constraints he has put into mind?

Input

Input will start with a line **T** containing the number of test cases. Each test case starts with **N**, the number of restaurants Rakib has rated. In the next line, there are **N** integers, where the **i'th** integer indicates the rating he has given for the **i'th** restaurant. The next line contains an integer **K**, the number of **categories** he wants to break the ratings into. This is followed by a line containing **K** integers, the **weight** of each **category**.

Output

For each test case output the case number followed by the answer. If there is no way to break the ratings into the categories maintaining the constraints given, output **-1** as the answer.

10^5

$1 \le A \le 10^9$

Otherwise, output the **minimum total weighted sum** of all the categories of all the ratings as the answer. (See sample for clarification)

Constraints:

1 <= T <= 22
1 <= N <= 100000
1 <= K <= 500
1 <= Rating for a restaurant <= 100000
1 <= Weight of each category <= 10
1 <= Points given for each category <= 10

Sample Input

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

Output for Sample Input

```
Case 1: 67
Case 2: -1
```

Explanation:

For each rating, a possible breakdown could be,

Original	Category 1 (Weight: 3)	Category 2 (Weight: 2)	Weighted Sum
1	1	1	5
2	3	1	11
2	1	4	11
3	1	5	13
3	3	2	13
4	2	4	14
		Total	67

Problem C

Rectangle Partitioning

You'll be given a rectangle of size **N** x **M**. Each of the cells of the rectangle can contain a blue lamp(‘**b**’) or a green lamp(’**g**’) or the cell is empty(‘**.**’).

Now you have to make exactly two partitions in the rectangle, one horizontally and the other one vertically. After these partitions, the rectangle will get divided into four non-empty sections. Each of the sections has a dissatisfaction value, which is the absolute difference between the number of blue lamps and green lamps in that section. And the total dissatisfaction of the rectangle is the summation of dissatisfaction values of all the four sections. Now you have to find an optimal partitioning of the rectangle such that the dissatisfaction value of the matrix gets minimized.

Input

The first line of the input contains an integer **T** which denotes the number of test cases.
The first line of each test case contains two integers **N** and **M.** Each of the N lines contains M characters representing the rectangle.

Constraints

- $1 \le T \le 20$
- $2 \le N, M \le 1000$
- **Possible characters inside the rectangle: [‘b’, ‘g’, ‘.’]**

Output

For each test case, print the answer in the format "**Case X: Y**", where Y is the minimum possible sum of dissatisfaction values after the optimal partitions.

Sample Input	Output for Sample Input
1 5 5 bb.gg ....b.. ..... ggbgg	Case 1: 4

Problem D

Yet Another Max Sum Problem

You are given an array **A** (1 based indexing) consisting of **N** integers. From each index **i** ($1 \le i \le N$) you can go to either **i + 1** or **i - 1** considering both of the indexes are valid. If you go to a certain index you have to take the integer in that index. You need to maximize the sum of taken integers. Though you can go to a specific index multiple times, you can't take the integer in that index more than once. Now you need to print **N** space-separated integers where the i^{th} integer denotes maximum sum including the integer in index **i**.

Input

The First line contains an integer T ($1 \le T \le 10$) denoting the number of test cases. Each test case starts with two integer N ($1 \le N \le 10^5$) denoting the number of elements in the array.

The next line contains **N** space-separated integers $A_1\ A_2\ A_3 \ldots A_N$ ($-10^9 \le A_i \le 10^9$).

Output

For each case first print the case number and then print **N** space-separated integers asked in the question. Please see the sample for details.

10^5

Sample Input	Output for Sample Input
2 3 5 10 1 3 5 -10 1	Case 1: 16 16 16 Case 2: 5 -4 1

$\boxed{10^9}$

5 10 1

~~5 15 16~~

1

1.	2	3
5	10	1

if i

$\boxed{10^5}$

10^5

10^9

10^9 = 10^9

10^2

0 5 10 i-1

11 1 0

11 6 10

5 15 -16

16 11 1

26 17

7

Problem E

Expected XOR

You will be given two arrays **A** and **B** both containing **N** non-negative integers. All elements of **A** are **distinct** and B_i* 10^{-5} means the probability of picking the i^{th} index in **A** if one picks blindfolded. First, you will randomly swap two different elements in **A** exactly once. Each pair of indexes are equally likely to be chosen in this step. Then you will make **K** moves. In each move, you will pick an index in **A** blindfolded according to the probability distribution in **B**. Moves are independent and you can pick the same index more than once. After **K** moves, your score is calculated by taking the bitwise XOR of the values in the **K** indexes that you picked from **A**. It can be shown that the expected score can be expressed in the form **P/Q** where **P** and **Q** are integers and coprime. Calculate $P*Q^{-1}$ modulo 10^9+7.

Input

There will be multiple test cases. The first line of input will contain the number of test cases **T** ($1 \le T \le 10$). For each test case, the first line contains two integers **N** and K ($2 \le N \le 10^5$, $1 \le K \le 10^9$). The next line contains **N** distinct integers A_1 A_2 A_3 ... A_N ($0 \le A_i \le 10^9$), denoting the elements in the array **A**. The following line contains **N** integers B_1 B_2 B_3 ... B_N ($0 \le B_i \le 10^5$), denoting the elements in the array B. It is guaranteed that the sum of all B_i is 10^5.

Output

For each test case, print a single line containing $P*Q^{-1}$ modulo 10^9+7. Please see the sample for details.

Sample Input

```
2
2 1
3 4
50000 50000
3 2
3 4 13
30000 20000 50000
```

Output for Sample Input

```
500000007
400000009
```

Problem G

Joker Strikes Again

Blenheim is a country of **N** cities. **There is a unique path between every pair of cities**.

Joker selects a city **U** randomly. Then he destroys exactly one of its adjacent roads randomly. After this, he selects another city **V** among the cities that can be reached from city **U** randomly. Joker can reach city **V** if there is a path from city **U** to **V** and that path doesn't contain the destroyed road. Find the expected distance from any city **U** to **V**. **U** and **V** can be the same. It can be shown that the expected distance can be expressed in the form **P/Q** where **P** and **Q** are integers and coprime. Calculate $P*Q^{-1}$ modulo 10^9+7.

Input:

- The first line of the input contains a single integer **T** denoting the number of test cases. The description of **T** test cases follows.
- The first line of each test case contains an integer **N**.
- Each of the next **N−1** lines contains two space-separated integers **x** and **y** denote that nodes **x** and **y** are connected by an edge.

Output:

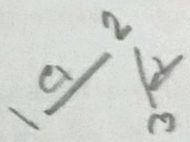
For each test case, print a single line containing the **case no** and the **expected distance** $D = (PQ^{-1}) \% MOD$. For more details, please see the sample I/O.
Where PQ^{-1} is the **expected distance** and $MOD = 10^9+7$.

Constraints:

- $1 \le T \le 10^4$
- $2 \le N \le 10^5$
- $2 \le \sum N \le 5*10^5$
- $1 \le x, y \le N$, $x \mathrel{!}= y$

Sample Input	Output for Sample Input
2 3 1 2 2 3 5 1 5 3 2 5 4 1 3	Case 1: 166666668 Case 2: 200000002

Explanation:



For the first test case :

The graph is like this :

1 -e_1- 2 -e_2- 3

Here,

nodes are 1, 2 and 3.

Edges are e_1 and e_2.

Now if we choose **1** as **U**, we have only one option, have to destroy the edge **e_1** and then choose **1** as **V**.

Similarly, if we choose **3** as **U**, we have only one option, have to destroy the edge **e_2** and then choose **3** as **V**.

If we choose **2** as **U**, we have two edges to destroy: **e_1** and **e_2**.

If we destroy **e_1**, then we can choose either **2** or **3** as **V**.

Similarly, if we destroy **e_2**, then we can choose either **1** or **2** as V.

So overall expected distance :

$$\tfrac{1}{3}*1*1*0 + \tfrac{1}{3}*1*1*0 + \tfrac{1}{3}*\left(\tfrac{1}{2}*\left(\tfrac{1}{2}*0 + \tfrac{1}{2}*1\right) + \tfrac{1}{2}*\left(\tfrac{1}{2}*0 + \tfrac{1}{2}*1\right)\right) = \tfrac{1}{6}$$

$$= 166666668 \text{ (mod by } 10^9+7)$$

Here, the probability of choosing each node is $\frac{1}{3}$.

For the first two scenarios as we have only one option, so the probability is 1. But here we don't find any positive distance, so their contribution is 0.

In the third scenario, we have two options in both choosing edges and nodes, so the probability is $\frac{1}{2}$.

we get positive distance 1 if choose edge e1 and node 3 or edge e2 and node 2.

Problem F

Maintain the Ratio

Given a binary string **S** of length **n** and an integer **k**.
Count the number of substrings **p** such that **Count(p, 0) >= k * Count(p, 1)**.
Here, Count(p, 0) denotes the count of 0 digits in **p**
and Count(p, 1) denotes the count of 1 digit in **p**.

There will be **q** independent test-cases.

Input

The first line will contain **q**, the number of test-cases.
Each of the next **q** lines will contain a string **S** and an integer **k**.

Constraints

- 1 <= **q** <= 10
- 1 <= **k** <= **n** <= 10^5

Output

For each test case, output in a single line the number of substrings maintaining the property.

Sample Input	Output for Sample Input
2 01001 2 00000 3	Case 1: 8 Case 2: 15

Problem H

Maximize AND Value

You are given a graph of **n** nodes and **m** edges. Each **i**th ($1 \le i \le m$) edge has a value, X_i assigned to it. You have to find an odd length cycle so that the beauty value of the cycle is maximized. **The graph doesn't contain any self-loops or multi-edges.**

The beauty value of a cycle is defined as the '**bitwise and**' between the weight of the edges on the cycle. More formally, if the edges on a k-length cycle are $E_1, E_2, E_3 \ldots E_k$ then the beauty value of this cycle = X_{E1} & X_{E2} & X_{E3} & … & X_{Ek}.

Input:

- First-line will contain t ($1 \le t \le 6$), number of test cases. Then the test cases follow.
- Each test case is formatted as follows:
  - First-line will contain two integers n($1 \le n \le 10^5$), number of nodes and m($0 \le m \le 10^5$) , number of edges.
  - Each of the next **m** lines will contain **three** integers: **u v X**, meaning that there is an edge between nodes **u** and **v** and the value assigned to this edge is X ($1 \le u, v \le n$, $0 \le X \le 10^9$).

Output:

For each test case, print the maximum beauty value possible which we can make from an odd length cycle from this graph in a single line. Please see the sample for more details.

Sample Input

```
2
5 6
1 2 7
1 3 14
2 4 6
4 5 6
3 5 7
3 4 5
6 7
1 2 2
1 3 4
2 4 2
3 4 1
3 5 6
5 6 7
6 4 2
```

Output for Sample Input

```
6
0
```

Explanation:

In the 1st graph, there are 2 cycles, 1->2->4->5->3->1 which has beauty value = 7 & 6 & 6 & 7 & 14 = 6,
another cycle is 3->4->5->3 which has beauty value = 5 & 6 & 7 = 4.So the answer is 6.

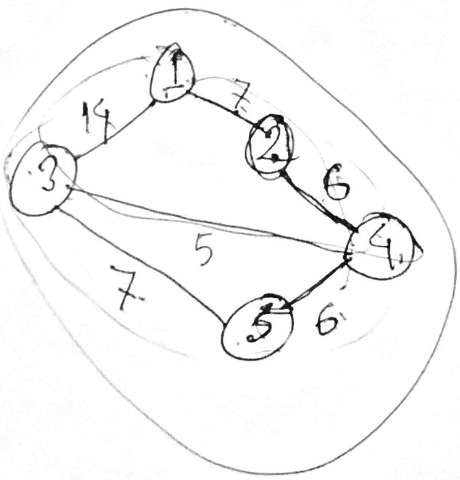
In the 2nd graph, there is no cycle. So the answer is 0.

7&6&6&7&14

1-2-4-3-5-1

1-2-4-5-3-1

3-4-5-3



Problem I

Special Set

You are given 2 integers **n k** .
We define a special set, **S** = {x : x is positive and **ncr(n, x)** is even}.
Find the **k**th smallest number in **S**.
Please note that elements in **S** are 1-indexed.

Input:

- First-line will contain **T (1 <= T <= 50,000)** number of test cases. Then the test cases follow.
- Each test case consists of a single line of input, two integers **n k**, as stated in the statement. **(1 <= n, k <= 10^{18})**

Output:

For each test case, output a single integer(answer to this test case) in a single line.

Sample Input

3 14 5 2 1 2 10	9 1 11

$ncr = \frac{n!}{r!\,(n-r)!}$

Problem J

XOR Attack

Petya is given an array **A**, consisting of **N** non-negative integers. He will append exactly one non-negative integer **D** in that array. Let's name this new array **B**.

He is given another integer **X**. He has to choose **D** in such a way that **D** is not present in **A** and **bitwise XOR** of all the elements of **B** is **X**.

You have to help him find **D**. If there are multiple answers, find the smallest of them. If there is no answer possible, print **-1** instead.

Input

Input starts with an integer, **T**, denoting the number of test cases. For each test case, the first line contains two integers, **N** and **X** denoting the size of array **A** and the given integer. The next line contains **N** space-separated integers denoting the elements of the array.

Output

For each test case print a single line in this format Case **T: D**.
Here **T** denotes the case number and **D** denotes the answer.

Constraints

1 <= T <= 100
1 <= N <= 100
0 <= A[i], X <= 1000000

Sample Input	Output for Sample Input
2 3 2 1 2 3 5 15 1 1 2 2 5	Case 1: -1 Case 2: 10

Problem K

Match the Strings

You will be given a string **S** of length **N** and another string **P** of length **M**.

Let's define a function, **F**(**X**, **Y**).

It takes two string **X** and **Y** having same length and returns an integer, their **match_score**. The **match_score** is total number of positions where the characters of both strings match. Example: **F**("**aia**", "**iai**") = **0**, **F**("**aia**", "**aaa**") = **2**, **F**("**abbadadda**", "**bbbaaaccc**") = **4**.

Print the summation of **F**(**X**, **P**) where **X** is a substring having length **M** of string **S**. So, there will be total **N-M+1** different **X**.

Input

Input starts with an integer, **T**, denoting the number of test cases. For each test case, the first line contains and string **S** and the second line contains another string **P**.

Output

For each test case, print a single line in this format Case T: D. Here **T** denotes the case number and **D** denotes the answer.

Constraints

1 <= T <= 100

1 <= M <= N <= 30000

The strings will contain only the first 10 lowercase English letters.

Sample Input	Output for Sample Input
2 abagig bgi haffidafidfid hafd	Case 1: 3 Case 2: 10