The International Collegiate Programming Contest
Sponsored by ICPC Foundation

EGYPTIAN COLLEGIATE
PROGRAMMING CONTEST

The 2020 Egyptian Collegiate Programming Qualification Contest

(Day 4 Contest Problems)

AAST
November 2020

Problem A. El-Khadrawy and the Array

Input file:	khadr.in
Output file:	standard output
Balloon Color:	Pink

El-Khadrawy has an array of N integers, he wants to make them all equals. He asked for your help.

You are allowed to make any number of operations (possibly zero). In each operation, you select an index i and a positive value X, and:

- add X to a_i (if a_i is odd)
- or subtract X from a_i (if a_i is even)

Each operation has a cost equal to the used X. El-Khadrawy wants to know, what is the minimum cost to make all elements equal?

Input

The first line of the input contains one integer T, the number of test cases.

The first line of each test case contains an integer N, $(1 \leq N \leq 2 \times 10^5)$, the number of elements in the array.

The second line of each test case contains N integers $a_1, a_2 ... a_N$ $(1 \leq a_i \leq N)$, the elements of the array.

Note that the sum of N over all test cases won't exceed 10^6.

Output

For each test case print one integer, the minimum cost to make all elements equals.

Example

khadr.in	standard output
1 7 1 2 2 3 3 3 7	14

Problem B. Jumping Game

Input file:	jump.in
Output file:	standard output
Balloon Color:	Yellow

You got bored from the contest that you decided to play a jumping game. In this game, you are given an array of size N. Each element inside the array is denoted as A_i.

You can make the game character start from any element in the array. However, from a cell i, the character can only jump to a cell j if $(j > i)$. The character likes to make their jumps between cells that have a difference equal to some power of two. In other words, a jump from cell i to cell j is possible if:

$$|Ai - Aj| = 2^k$$

Where $|x|$ denotes the absolute value of x, and k is some non-negative integer.

A sequence of jumps that starts from some cells inside the array and ends at some cell is called a trip. You want to count the number of different trips that the character can make. Two trips are considered different if they differ by at least one cell (The difference is considered based on the cell's index, not value).

Input

The first line contains a single integer T denoting the number of test cases.

Each test case starts with a line containing N $(1 \leq N \leq 10^5)$ denoting the length of the array.

The next line contains N integers A_i $(0 \leq A_i < 2^{20})$ denoting the elements of the array.

Output

For each test case print a single line containing a single integer denoting the number of trips that the character can make, modulo $(10^9 + 7)$.

Example

jump.in	standard output
2 5 1 9 5 10 11 6 8 10 2 7 9 18	16 20

Problem C. Bus Stations

Input file:	bus.in
Output file:	standard output
Balloon Color:	Black

The street in which **El-Khadrawy** lives contains N bus stations on a line numbered from 1 to N. The number of buses that serve this street is also N. El-Khadrawy lives near the U^{th} bus station. On minute T_s El-Khadrawy will be standing at the bus station near his home because he will be late for his date which is near the V^{th} bus station.

El-Khadrawy knows the schedule of the buses. On the moment B_i, the i^{th} bus starts from either the first bus station with number 1, or the last bus station with number N. All buses move at a constant speed of one bus station per minute. Once a bus starts working on minute B_i it will work infinitely.

If a bus starts at the first bus station, its route will be $1, 2, .., N-1, N, N-1, .., 2, 1, 2, ..$ and so on.

If a bus starts at the last bus station, its route will be $N, N-1, .., 2, 1, 2, .., N-1, N, N-1, ..$ and so on.

El-Khadrawy is a lazy guy, so he will surely not walk. He will not change buses either. He will just get on a bus that reaches station U at some minute not smaller than T_s, and then get off on bus station number V. He is interested in calculating the earliest minute in which he can be at the station V.

Input

The first line contains a single integer T indicating the number of test cases.

Each test case starts with a line containing four integers N, U, V, and T_s $(1 \leq U, V \leq N \leq 10^5)$ $(U \neq V)$ $(N \geq 2)$ $(1 \leq T_s \leq 10^9)$ indicating the number of bus stations (which is the same as the number of buses), the starting and ending bus stations and the minute in which El-Khadrawy will be standing next to the U^{th} bus stations. It is guaranteed that the sum of N over all test cases does not exceed 3×10^5.

N lines follow. Each line contains two integers B_i and S_i $(1 \leq B_i \leq 10^9)$ $(0 \leq S_i \leq 1)$. If S_i equals to zero, it means the i^{th} bus will start from the first bus station at minute B_i. If S_i equals to one, then it starts from the last bus station at minute B_i.

Output

For each test case print a single line containing a single integer, denoting the earliest moment in which El-Khadrawy can be at the station V.

Example

bus.in	standard output
1 3 2 3 5 1 0 2 1 3 0	6

Problem D. Contests

Input file:	`contests.in`
Output file:	`standard output`
Balloon Color:	`Cyan`

El-Khadrawy is a contestant in the collegiate programming contests. He joined the contests hoping to become among the greatest contestants. However, since he is a little low on cash this year, he discovered that he needs to collect money to pay his college fees as he must be in college to participate in the contests.

To collect the money he has two options.

He can either create problems and sell them for X pounds each. Or, he can sell newspapers in the street and get Y pounds for each sold newspaper. The funny thing is that the time needed to create a problem turned out to be exactly similar to the time needed to sell a newspaper.

Now, he has few days left before the season starts and he needs your help to find the best strategy to collect the maximum amount of money. El-Khadrawy can only do one job. Should he create problems or sell newspapers?

Input

The first line contains a single integer T denoting the number of test cases.

Each test case consists of a line containing two integers X and Y $(1 \leq X \leq 20)$ $(100 \leq Y \leq 500)$ denoting the number of pounds that El-Khadrawy gets from creating a problem and from selling a newspaper, respectively.

Output

For each test case print *Problem*, if El-Khadrawy should create problems or *Newspaper* if he should sell newspapers.

Example

contests.in	standard output
1 2 152	Newspaper

Problem E. Flower Tree

Input file:	`smell.in`
Output file:	`standard output`
Balloon Color:	`Orange`

Speed land consists of N cities and $N-1$ roads such that each pair of cities has a path between them.

El-Khadrawy Group (El-G for short) has made a spell that generates an outstanding flower smell. Then, the group will pick a single city (it could be any city) as the starting source of the flower smell. Each day, the flower smell spreads randomly to a single city that is near an already smelling city and didn't have the smell before. After N days, all cities of speed land will be smelling from El-G's flowers.

El-G is keeping track of the spread. In the beginning, they have an empty list. Every day, when a new city starts to smell like a flower, they add it to the end of their list. Of course, the first city to be added is a random city. However, the next cities have to be adjacent to an already smelling one.

El-G asks you to count the number of different scenarios such that all cities have an outstanding smell from his spell. In other words, they ask you to count the number of different lists that they might have. Since the answer can be large, find it modulo $10^9 + 7$.

Two scenarios are different if in the first one, city X starts to smell before the city Y, and in the second one, city X starts to smell after the city Y.

Input

The first line of the input contains one integer T, the number of test cases. Then T test cases follow.

The first line of each test case contains one integer N ($1 \le N \le 10^5$) denoting the number of cities.

The next $N-1$ lines describe the roads. The i^{th} road is given as two integers u_i, v_i ($1 \le u_i, v_i \le N$), where u_i and v_i are the cities that the i^{th} road connects. It is guaranteed that there exists a path between every pair of cities. So, speed land is connected. It is also guaranteed that there is at most one road between every pair of cities.

It is guaranteed that the sum of N does not exceed 2×10^6, $\sum_{i=1}^{T} N_i \le 2 \times 10^6$.

Output

For each test case, print the answer: the number of ways such that all cities of speed land start to smell with El-G's flowers modulo $10^9 + 7$.

Example

smell.in	standard output
1 7 1 2 1 3 2 4 3 5 3 6 6 7	164

Example

ninja.in	standard output
2 5 2 1 3 3 5 4 6 1 3 1 5 0 2 0 4 10 5 1 15 25 30 2 17 15 20 16 30 12 15 11 11 11 11 12 12 12 12 0 9 2 7 1 9 2 8 3 3	1 2 5 2 4 3 0

Problem F. Ninja Floppy Bird

Input file:	`ninja.in`
Output file:	`standard output`
Balloon Color:	White

You are given an N points on the x-coordinate axis numbered from 0 to $N-1$. In addition, you are given N pairs (L_i, R_i). Each of these pairs can be considered as a range on the y-coordinates axis $[L_i, R_i]$. Therefore, each point on the x-coordinate axis has its own range on the y-coordinate axis.

Suppose you have a floppy bird that is at point $x = i$, then the bird can only cross this x if his y-coordinate value is inside the range $[L_i, R_i]$. When moving from one point to the other over the x-axis, you need to change the bird's y coordinate to be able to pass him through the ranges $[L_i, R_i]$.

As a result, you are given Q queries. Each query consists of two values A and B. You are asked to start your floppy bird at the a^{th} point on the x-axis and let him fly until it reaches the b^{th} points on the x-axis. The bird can change its y-axis coordinate at a cost of 1 to any other value. In other words, the bird needs to pass the range $[A, B]$ starting from A and ending at B. What is the minimum number of times that the bird needs to change its y-axis coordinates?

Note that the bird always flies on a horizontal line parallel to the x-axis. Therefore, if the bird's current y-axis coordinate is Y, then it passes through the i^{th} point if:

$$L_i \leq Y \leq R_i$$

In addition, the bird can pick its initial y-coordinate value before flying over the a^{th} point at no cost.

Input

The first line contains a single integer T denoting the number of test cases.

The second line contains two integers N and Q ($1 \leq N, Q \leq 10^5$) denoting the number of points and the number of queries.

Then, N lines follow - each of them contains two integers L_i and R_i ($1 \leq L_i \leq R_i \leq 10^9$) denoting the y-coordinates range where the bird can pass on the i^{th} point.

The next Q lines contain the queries. Each line contains two integers A and B ($0 \leq A \leq B \leq N-1$) denoting the range of the points that the floppy bird has to fly through.

Output

For each query print a single line containing a single integer - the answer to the query.

Problem G. AboAdnan's Dilemma

Input file:	deli.in
Output file:	standard output
Balloon Color:	Blue

AboAdnan was preparing for a contest. Each day, he will solve several problems, and he will write the number of solved problems on a piece of paper. Then, he will give this paper to his coach Resli to evaluate his progress.

AboAdnan spent N days solving problems. Therefore, he wrote N values $a_1, a_2, ..., a_n$, where a_i represents the number of problems that he solved on the i_{th} day. He was going to give this paper to his coach, when he remembered that the coach gets mad if he notices that AboAdnan was solving fewer problems each day for more than K consecutive days.

In other words, coach Resli gets mad if he sees a sub-array of length more than K, such that the elements in this sub-array are strictly decreasing in value. For example, arrays $[10, 9]$, $[6, 2, 1]$ and $[4]$ are strictly decreasing, but arrays $[4, 4]$ and $[3, 4, 6]$ are not.

Well, AboAdnan doesn't want to see his coach getting mad. So, he will remove the minimum number of elements such that the remaining array doesn't contain any sub-array of length more than K with strictly decreasing values. Note that AboAdnan can't change the order of the elements, or otherwise his coach will suspect that he is cheating.

AboAdnan asked the judges of this contest to help him, but no one was able to solve this task. If you don't help AboAdnan, his coach won't let him participate in the next contest.

Input

The first line of the input contains one integer T, the number of test cases.

The first line of each test case contains two integers N and K, where $(0 \leq K \leq N \leq 1000)$ and $(N \geq 1)$.

The second line of each test case contains N integers $a_1, a_2, ..., a_n$ $(-10^9 \leq a_i \leq 10^9)$, where a_i is the number of problems that aboAdnan solved in the i^{th} day.

The sum of N over all test cases will not exceed 10^4.

Output

On the first line of each test case, print one integer M, the minimum number of elements that AboAdnan should delete such that his coach won't get mad.

On the second line of each test case, print $N - M$ space-separated integers, the remaining elements in the array. If there are multiple solutions, you can print any of them.

Example

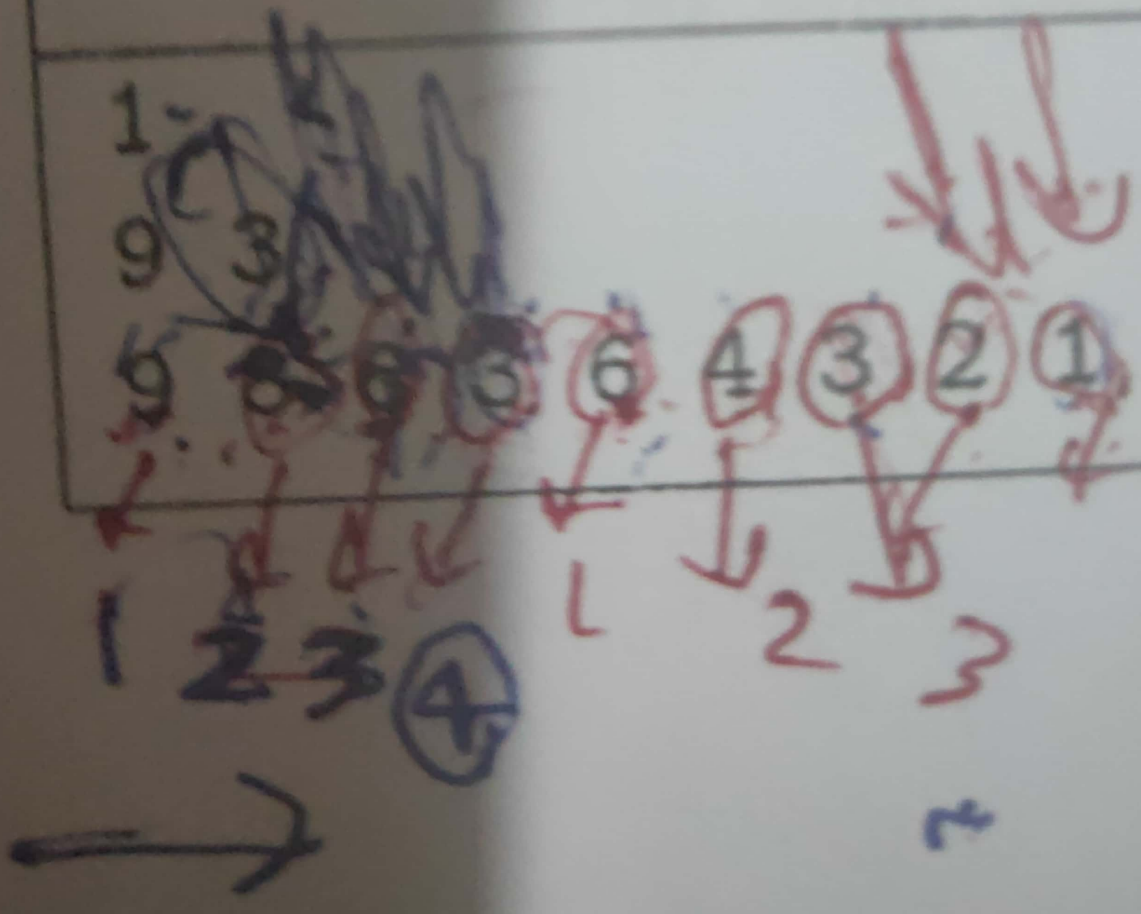
deli.in	standard output
1 9 3 9 8 6 6 4 3 2 4	3 9 8 6 6 4 3

On the first line of each test
should delete such that his

On the second line of each t
array. If there are multiple

Example

deli.in



Problem H. Eating Letters

Input file:	eat.in
Output file:	standard output
Balloon Color:	Gold

You are given two strings, S of length n, and P of length m, of lowercase English letters. **The letters of P are distinct.**

You can do the following operation, zero or more number of times:

Choose a letter c that exists in both strings, S and P, then either delete all the occurrences of c in s, or leave them all.

You are asked to print the lexicographically minimum string you can get from S, by applying the described operation, zero or more number of times.

A string a is lexicographically smaller than a string b, if a is not equal to b, and one of the following conditions is met:

- a is a prefix of b
- a is not a prefix of b, and there is a position j, such that, for each $i < j$, $a_i = b_i$, and $a_j < b_j$.

Obviously, the empty string is lexicographically the smallest one.

A string a is a prefix of a string b, if $|a| \leq |b|$, and the string obtained from the first $|a|$ letters of b is equal to a. Where $|S|$ denotes the length of the string S.

Input

In the first line of the input, you are given T, the number of test cases.

In the first line of each test case, you are given n and m, where $(1 \leq n \leq 10^5)$, and $(1 \leq m \leq 26)$.

In the second line of each test case, you are given a string of length n, of lowercase English letters, that is S.

In the third line of each test case, you are given a string of length m, of distinct lowercase English letters, that is P.

The sum of n over all test cases doesn't exceed 10^5.

Output

For each test case, print two lines.

In the first line, print the length of the lexicographically minimum string you can get from s.

In the second line, print that string. If the string is empty, then print an empty line.

...mple

eat.in	standard output
3 10 3 codeforces ade 4 4 spoj spoj 9 2 cplusplus lp	10 codeforces 0 7 cluslus

Problem I. El-Khadrawy and Bots

Input file:	`bots.in`
Output file:	`standard output`
Balloon Color:	`Green`

El-Khadrawy is the red king in a faraway land. He spent his life making his students the best ICPCians in the world.

The last wizard in this century is always trying to write retirement spells that make all ICPCians quit training.

Since El-Khadrawy wants all students to keep training for ICPC, he decided to make a bot that will break all the retirement spells.

In order to check if the given spell S is a retirement spell, the bot will check if there is a string T equal to "*retire*" as a subsequence of the spell S.

El-Khadrawy wants the bot to display **Not yet** If the given spell S is a retirement spell. Otherwise, the last wizard will tell the bot **You lost**.

Input

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

Each test case contains two lines. The first line contains one integer N ($1 \leq N \leq 10^5$), while the second line contains the string S that consists of N lowercase Latin letters denoting the spell that El-Khadrawy has to check.

Output

For each test case, if S is a retirement spell, then print: *Not yet.*

Otherwise, print: *You lost.*

Example

bots.in	standard output
2 10 retirement 6 return	Not yet You lost

Note

A subsequence of a string S can be obtained by removing zero or more characters from S.

Problem J. Cellular Geometry

Input file:	`cellular.in`
Output file:	`standard output`
Balloon Color:	`Silver`

In cellular geometry, every area is divided into clusters, and each cluster is divided into some number N of cells. The value of N affects many things: number of users, antenna transmitting power, interference with other cells, mobile battery life, and so on.

N is calculated by the equitation:

$$N = i^2 + i * j + j^2$$

Where $i \geq 0$, $j \geq 0$ and $i + j > 0$.

It's clear that N can't be equal to some values, for example there is no cluster with size 6 or 10.

Now given N you should decide whether there is a cluster of size N.

Input

The first line will contain t indicating the number of test cases.

Each test case contains one line with a single integer N ($0 \leq N \leq 1000$).

Output

For each test case print one line, *YES* if the is a cluster of size N, and *NO* if there isn't.

Example

cellular.in	standard output
4	NO
5	YES
7	YES
9	NO
14	