Problem A. Can Shahhoud Solve it?

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Pink`

Shahhoud is participating in a programming contest, he only needs to solve one problem to qualify to the next round.

Shahhoud is so experienced that he can find out the difficulty of any problem as soon as he reads it. He found out that the easiest problem in the contest has a difficulty of D.

Shahhoud's problem solving skill is S, and he knows that he can solve the problem if and only if his skill is higher or equal to the problem's difficulty (i.e. $S \geq D$). Can you tell Shahhoud Whether or not he can solve the problem?

Input

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

Each test case consists of a single line containing 2 space-separated integers, S and D. ($1 \leq S, D \leq 100$) denoting Shahhoud's skill, and the problem's difficulty respectively.

Output

For each test case, print one line containing a single word, "YES"if Shahhoud can solve the problem, and "NO"otherwise.

Example

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

Problem B. Defeat the Monsters

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Yellow`

Joud likes playing video games so much. Recently he found a game that can increase his skills in both video games and mathematics.

Inside this weird video game Joud needed to fight N monsters, where the i^{th} monster has power A_i. The problem in this video game is that you have to use your mathematical skills to defeat the monsters using the **minimum** number of steps. Each monster is defeated when its power reaches 1. Sarcastically some monsters can start the game dead (i.e. with a power equals to 1).

In each step Joud can choose **at most** 3 different monsters with indices i, j and k. After that Joud chooses any prime number ($P \geq 2$) he wants and performs the following operations simultaneously:

1. $A_i = A_i \ / \ P^u$, where $(0 \leq u \leq 64)$ and $(A_i \bmod P^u = 0)$.
2. $A_j = A_j \ / \ P^v$, where $(0 \leq v \leq 64)$ and $(A_j \bmod P^v = 0)$.
3. $A_k = A_k \ / \ P^w$, where $(0 \leq w \leq 64)$ and $(A_k \bmod P^w = 0)$.

The game stops when all the monsters are defeated. Can you determine what is the minimum number of steps Joud needs to finish the video game?

Input

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

Each test case start with a line containing an integer N $(1 \leq N \leq 10^5)$, denoting the number of monsters.

The next line contains N space-separated integers A_i $(1 \leq A_i \leq 10^6)$ denoting the power of the i^{th} monster.

Output

For each test case print a single line containing one integer, the minimum number of steps needed to defeat all monsters.

Example

standard input	standard output
4 3 2 4 8 4 4 8 6 6 5 1 2 4 8 16 5 30 30 6 10 15	1 3 2 6

Problem C. UCL Game Night

Input file:	standard input
Output file:	standard output
Balloon Color:	White

Saeed is planning to watch the UEFA Champions League game tonight. Sadly, not all the pixels in the TV screen are working, and some of them are just black non-working dots!

Saeed decided not to give up, and focus his eyes on a rectangular area of the total TV screen. This area might have bad pixels, that's why Saeed decided that he would choose a rectangle that has no more than K black non-working pixels.

TV screen can be represented as an NxM grid, containing pixels P_{ij}. Each pixel either equals to 0 indicating a working pixel, or 1 indicating a black non-working pixel. The rectangle's height and width that Saeed is going to choose should be parallel to the X and Y axes.

Saeed is interested in the area of the largest rectangle he can choose, such that it doesn't contain more than K bad pixels. He couldn't find an algorithm to calculate such area, so he decided to make this a problem in TCPC 2018, maybe you guys can help him out!

Input

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

Each test case starts with a line containing three space separated integers N, M and K $(1 \leq N, M \leq 100)$ $(1 \leq K \leq 10000)$ denoting the number of rows and columns of the TV, and the maximum number of bad pixels Saeed can stand inside his chosen rectangle.

The next N lines contain M space-separated integers each P_{ij} $(0 \leq P_{ij} \leq 1)$ indicating the status of the pixel at line i and column j.

Output

For each test case print one line containing a single integer, the maximum area of a rectangle that Saeed can choose.

Example

standard input	standard output
3 3 3 2 1 0 1 0 0 0 1 0 0 4 4 2 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 5 5 5 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 0 1 1 1	6 9 16

Problem D. Police Stations

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark Red`

Ahmad was assigned as the new road engineer for the city. His first task was to install police stations along the road connecting the center of the city and the governor house.

The governor suggested to Ahmad N points along the road were the government can build police stations. Ahmad needs to choose at most K points to build police stations in them. The i^{th} suggested police station has the following information about it:

1. P_i The distance of this police station from the center of the city.

2. A_i When the i^{th} suggested police station is chosen to be built, no other police station can be chosen in the range [P_i - A_i, P_i + A_i] inclusive. Also no two ranges [P_i - A_i , P_i + A_i] and [P_j - A_j , P_j + A_j] can intersect for any two chosen police stations i and j.

3. C_i If the i^{th} police station is chosen to be built, it increases the security of the road by C_i.

4. X_i Whenever two stations are chosen to be built, if no stations between them are chosen, the absolute difference of their security values must be at least X_i. In other words $|C_i$ - $C_j| \geq X_i$, Where $P_i > P_j$ such that Pi and Pj are consecutive chosen stations.

Ahmad got lost with all the values and parameters given to him by the governor, but he is sure of one thing: The governor wants this road to be as secured as possible. Write a program that would choose the optimal number of stations such that the overall security obtained is **maximized**.

Input

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

Each test case starts with two space-separated integers N and K ($1 \leq N \leq 10000$) ($1 \leq K \leq 100$) indicating the number of suggested police stations, and the maximum number of police stations Ahmad can choose.

Next follow the specification for police stations:

The second line of each test case contains N space separated integers P_i ($1 \leq P_i \leq 10^9$).

The third line of each test case contains N space separated integers A_i ($1 \leq A_i \leq 10^9$).

The fourth line of each test case contains N space separated integers C_i ($1 \leq C_i \leq 10^9$).

The fifht line of each test case contains N space separated integers X_i ($1 \leq X_i \leq 10^9$).

Output

For each test case, print one line containing a single integer, indicating the maximum security Ahmad can achieve by choosing the police stations to be installed optimally.

Example

standard input	standard output
1 3 2 1 5 9 3 3 4 1 3 6 2 2 2	7

Problem E. Create Your Own Nim Game

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Orange`

It is said that the best problems are the ones that both teach you and test you. Well, luckily for you the judges have decided to present such a problem in this contest!

Nim Game is a well-known game. The rules of the game are actually simple. The game consists of M piles and N stones. Each pile has a number of stones inside it, and each stone belongs to exactly one pile. Two players play the game alternately. In each step the player chooses a pile, and removes as many stones from it as he wishes. The player whose turn comes while all the piles are empty loses the game. Tricky game huh?

It can be proven that the winner can be determined before the start of the game, assuming both players play optimally. To determine the winner you must first calculate the logical XOR operator for all piles sizes, let's denote this value as X.

$X = S_1 \ xor \ S_2 \ xor \ ... \ xor \ S_M$

Where S_i denotes the size of the i^{th} pile. After that you just need to examine the value X. If X equals to 0 then the player to start **second** can always find a way to win, otherwise the player to start **first** can always find a way to win.

Now after reading this beautiful story what is your job? Well, your job is that given an integer N, find the **number of valid nim games** that contain exactly N stones, and the player to start **first** is guaranteed to win. In other words the sum of numbers of stones inside all piles must equal to N, and the player to play first must be the winner at the end of the game (i.e. X must be equal to a non-zero integer). Two ways are considered different if either M is different, or the number of stones in the i^{th} pile differ. For example "1, 2, 3"and "1, 3, 2"are considered two different ways. Since the answer can be large, print it modulo $10^9 + 7$.

Input

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

Each test case contains a single integer N ($1 \leq N \leq 100$) indicating the number of stones from which your nim game must consist.

Output

For each test case, print one line containing a single integer, the answer to the problem.

Example

standard input	standard output
2 1 3	1 4

Note

In C++, Java and most programming languages the XOR operator is denoted as $\wedge$.

Problem F. Nim Cheater

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Purple`

The judges really like nim game, so they decided to put yet another nim game problem into the problem set (sorry for that in advance).

Nim Game is a well-known game. The rules of the game are actually simple. The game consists of M piles and N stones. Each pile has a number of stones inside it, and each stone belongs to exactly one pile. Two players play the game alternately. In each step the player chooses a pile, and removes as many stones from it as he wishes. The player whose turn comes while all the piles are empty loses the game. Tricky game huh?

It can be proven that the winner can be determined before the start of the game, assuming both players play optimally. To determine the winner you must first calculate the logical XOR operator for all piles sizes, let's denote this value as X.

$X = S_1 \ xor \ S_2 \ xor \ ... \ xor \ S_M$

Where S_i denotes the size of the i^{th} pile. After that you just need to examine the value X. If X equals to 0 then the player to start **second** can always find a way to win, otherwise the player to start **first** can always find a way to win.

After learning this great game you decided to play it with your friend after this contest is finished. You arranged some piles, placed a number of stones inside each pile and decided that you would play **first**. Before starting the game your friend got busy, and you were given a chance of being alone with the piles for a few seconds!

You decided to guarantee your win. You have only a few seconds before your friend returns, so the only action you can do is to remove some (possibly zero) stones from some (possibly zero) piles, and throw them away. Since you don't want to be caught you want to do so as fast as possible. You are asked to determine the minimum number of stones you need to remove.

Input

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

The first line of each test case contains a single integer M ($1 \leq M \leq 10^5$), indicating the number of piles.

The next line contains M space-separated integers S_i ($1 \leq S_i \leq 10^9$), the number of stones in the i^{th} pile.

Output

For each test case print one line containing a single integer, the minimum amount of stones you should remove so that you can win the game. It is guaranteed that the answer exists and you can guarantee your win after removing some (possibly zero) stones.

Example

standard input	standard output
2 2 2 3 3 2 3 1	0 1

Problem G. Colors Overflow

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark Green`

If there is one thing Ziad is good at, it's creating new colors. In order to use his magical skill to its limit, he decided to start a new career in painting. In order to do so he asked Joud to help him.

Joud presented Ziad with an infinite length line. Initially all points of the line are colorless. After examining the line, Ziad noticed that it's not an ordinary line. Even if a point is painted multiple times, it keeps all the colors without mixing them, and when looking at it you can clearly see all the colors separated.

In order to test Ziad's skills Joud presented him also with Q queries. Each query is of the type "t L R". Queries can be of the following two types, depending on the value of t:

1. "1 L R"This is a task for Ziad to color each sub-segment of the range [L, R] inclusive, each with a unique color that has never been used before. For example when asked to paint the range [3, 6] he will paint segments [3 3], [3 4], [3 5], [3 6], [4 4], [4 5], [4 6], [5, 5], [5 6] and [6 6], each of them with a unique color that has never been used before.

2. "2 L R"This query means Ziad has to look at the range [L, R] inclusive, and write down the number of different colors that appear inside the range.

Anyways, Ziad is a little busy packing sugar and tea for his trip to be a judge, so he asked for your help answering these queries.

Input

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

Each test case start with a line containing an integer Q ($1 \leq Q \leq 10^5$), denoting the number of queries.

The next line contains Q lines describe the queries. Each line has one query of the form "t L R"($1 \leq t \leq 2$) ($1 \leq L \leq R \leq 10^5$).

Output

For each test case and for each query of the second type, print a single line, indicating the number of unique colors in the range [L, R].

Example

standard input	standard output
1 3 1 1 4 2 1 2 2 2 4	7 9

Problem H. Don't Ever Ask a Girl for her Codeforces Account

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Foshia`

While preparing teams for the upcoming ACM contests, coach Ramzi had a strange incident!

One day a girl asked coach Ramzi for his help regarding a problem. Coach Ramzi, as innocent as he is, asked the girl for her codeforces account. Suddenly the chat screen became blur! Yes the girl blocked him!!

Next day coach Ramzi wanted to go out, but then he realized, what if that girl saw him on the road? Just imagine what she can do to him after he asked for her codeforces account!

The country at which coach Ramzi lives can be represented as a tree with N nodes numbered from 1 to N. Coach Ramzi lives at city A, while the girl lives at city B. Coach Ramzi want to calculate the number of cities C and D, such that if he goes to city C, and the girl goes to city D, their paths won't intersect at any point.

Coach Ramzi is still shocked over what happened, and he can't get his mind straight. Could you write a program to answer this problem?

Input

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

Each test case starts with a line containing three space-separated integers N, A and B ($1 \leq N \leq 10^5$) ($1 \leq A, B \leq N$), denoting the number of cities, and the cities were coach Ramzi and the girl live respectively.

The next $N-1$ lines contain two space-separated integers each a_i b_i indicating that a road exists between city number a_i and city number b_i.

Output

For each test case print a single line containing one integer, the answer to the problem.

Example

standard input	standard output
1 5 1 3 1 2 2 3 2 4 4 5	7

Problem I. Odd and Even Queries

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Gold`

Once Daniar was explaining multiplication for his little brother from a mathematical book. At the end of the book there was an exercise labeled "For smart people only". Daniar got excited and decided to solve the exercise no matter what.

In the exercise Daniar was given an array A consisting of N positive integers. After that there were many questions. Each question had the following format "L R t". Questions were of 2 types depending on the value of t:

1. "L R 0"asking for the number of sub-sequences whose elements are taken from indices between L and R inclusive, and their multiplication result is even.

2. "L R 1"asking for the number of sub-sequences whose elements are taken from indices between L and R inclusive, and their multiplication result is odd.

Please note that a sub-sequence can be obtained of the original array by choosing one or more elements, without changing the relative order of their indices.

Daniar needs some time to think about a solution for this problem, but he doesn't want to look puzzled in front of his brother. Could you solve this problem and help your friend Daniar? Since the answer can be very large, print it modulo $10^9 + 7$.

Input

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

Each test case starts with two space-separated integers N and Q ($1 \leq N, Q \leq 10^5$) indicating the number of array elements and the number of questions respectively.

The next line contains N space separated integers A_i ($1 \leq A_i \leq 10^9$) indicating the i^{th} element.

The next Q lines represent the questions. The i^{th} question is "L_i R_i t_i "($0 \leq t_i \leq 1$) ($1 \leq L_i \leq R_i \leq N$) denoting the the left, right indices limits and the type of the query as explained in the statements.

Output

For each test case, print Q lines. The i^{th} line must contain the answer to the i^{th} question.

Example

standard input	standard output
1 5 3 7 11 12 2 5 1 3 1 1 2 0 3 4 1	3 0 0

Problem J. Weird Sum

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Black`

Whenever Saeed proposes a solution to a problem, Shahhoud tells him that it is incorrect and doesn't let him submit it.

This has happened so many times (like the famous BFS incident).

Saeed came up with a plan to prevent this. Whenever Saeed wants to submit a solution, he gives Shahhoud a problem to distract him while he submits.

This time, he gave him an array A of N numbers, and asked him to find the following sum:

For each pair (i, j) such that $i < j$. Let M be the maximum number in the array between i and j i.e. $M = max(A_i, A_{i+1}, ..., A_{j-1}, A_j)$. If the xor of A_i and A_j is greater than M, add M to the sum.

Can you help Shahhoud solve the problem before Saeed can submit another problem?

Input

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

Each test case start with a line containing an integer N ($1 \leq N \leq 10^5$), denoting the length of the array A.

The next line contains N space-separated integers A_i ($1 \leq A_i \leq 10^6$).

Output

For each test case print a single line containing one integer, the required sum.

Example

standard input	standard output
1 5 3 8 10 6 1	44

Problem K. Quantum Stones

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark Blue`

Nasser is one of the best quantum scientists in the world. Recently, he made a discovery that will revolutionize transportation.

After years of non-classic programming, he created Quantum Stones. These stones, when applied to a road, allow you to choose any prime number y that is a divisor of the road's cost and divide the road's cost by it. However, activating the stone costs twice the chosen prime number $(2 * y)$. **Each road can be affected by one stone at most**.

In order to show the importance of these quantum stones, he wants to show how greatly his invention reduces the cost of travelling between cities in his new paper called "Non-classic programming is the future".

However, finding the travelling cost between two cities is considered classic programming, so he doesn't want to waste time doing it. He asked you to help him.

There are n cities numbered from 1 to n and m bidirectional roads in Quantumia. The i^{th} road has a cost of w_i, and connects cities u_i and v_i.

Nasser has K quantum stones that he can use to reduce the travelling cost between city number S and city number E. What is the minimum cost after activating K stones at most?

Input

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

The first line of each test case contains three space-separated integers n , m , $k(1 \leq n, m \leq 10^4)$, $(1 \leq k \leq 100)$ the number of cities and the number of roads and the number of quantum stones respectively.

Each of the following m lines each contains three space-separated integers $(1 \leq u_i, v_i \leq n)$, $(1 \leq w_i \leq 10^6)$, this mean that there is a road with a cost of w_i between cities u_i and v_i

The following line contains two integers S , E $(1 \leq S, E \leq n)$, the starting and ending cities.

Output

For each test case, print one line containing one integer, the minimum cost to travel between S and E after applying the stones.

if there is no answer print -1 .

Example

standard input	standard output
3 3 4 2 1 2 15 1 3 20 2 3 24 1 3 18 1 3 4 4 1 1 2 18 2 3 64 3 4 42 1 3 36 1 4 4 4 10 1 2 2 1 3 5 2 3 11 3 4 13 1 4	12 56 18

Problem L. Odd and Even Count

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light Blue`

Recently, Riad decided to participate in ACM. Since he wants to qualify to ACM ICPC as soon as possible, he decided to solve hard problems.

The first problem was to count the number of odd numbers and the number of even numbers between 1 and N.

Riad is struggling with this problem, can you help him?

Input

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

Each of the following T lines contains a single integer N ($1 \leq N \leq 10^9$).

Output

For each test case, print a line containing two space-separated integers, the first one is the number of **odd** numbers between 1 and N , and the second one is the number of **even** numbers between 1 and N.

Example

standard input	standard output
2 5 6	3 2 3 3