Problems Overview

Problem A: Scoreboard

Problem B: Network Upgrade

Problem C: Proper Divisor

Problem D: Gem Mining

Problem E: Meeting Point

Problem F: Cipher

Problem G: Object Clustering

Problem H: Welcome

Problem I: Geeky Fun Fact

Problem J: Number Game

Note: The input and output for all the problems are standard input and output.

Problem A: Scoreboard

In an online programming contest, the scores and ranks of all teams are updated live. Teams are displayed in non-ascending order of their scores. The rank of a team is one plus the number of teams having scores strictly higher than its score. For example, given the following scores list:

```
100 90 90 80
```

the corresponding ranks would be

```
1 2 2 4
```

Given the current scores of the all the teams, your task is to find the most popular rank (the one with the highest number of teams). If there is more than one answer, you should take the lowest rank.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

Each data set consists of two lines where the first line contains an integer n ($n \leq 100$) representing the number of teams. The second line contains n space-separated integers representing the scores of n teams, not necessarily in a sorted order (the scores are at most 10^5).

Output

For each data set, write out on one line the lowest most popular rank.

Sample Input	Sample Output
2 4 100 90 90 80 5 5 4 3 2 1	2 1

Problem B: Network Upgrade

Reboot has a computer network consisting of n computers interconnected by m bi-directional information transmitting channels between pairs of computers. The computers are numbered from 1 to n; the channels are numbered from 1 to m. With the current network, two computers can exchange information directly to each other via the channel connected between them, or indirectly via intermediate computers. It is possible that there are pairs of computers that cannot exchange information to each other.

As the business is expanding, the company's director would like to set up a project to increase the bandwidth of some existing channels. While a channel is being upgraded, this channel cannot be used to transmit information. Therefore, during the time the upgrade project is being executed, some pairs of computers might not be able to exchange information anymore. In order not to interfere with the information exchange capability between computers in the company, the director is interested in a project that would upgrade the *largest* number of channels in the network while maintaining the following requirement: if a pair of computers can exchange information between each other, they would still be able to exchange information while the project is being executed.

Your task is to help the Reboot's director to determine the number of possible projects satisfying the stated requirement.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

The first line of each data set contains two positive integers n, m ($n \le 500$; $m \le 50000$). The i^{th} line in the next m lines contains two positive integers u_i, v_i indicating the indices of two computers connected by the i^{th} channel, $i = 1, 2, ..., m$.

Two consecutive numbers on the same line are separated by at least a space.

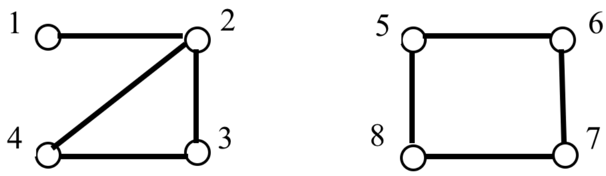
Output

For each data set, write in one line the remainder of the possible projects number divided by 10^9+7.

Sample Input	Sample Output
1 8 8	12

1 2 2 3 2 4 3 4 5 6 6 7 7 8 8 5	

Explanation: The following diagram illustrates the current computer network connections at Reboot.



With the current network, the computer numbered 1 *can* exchange information with the computers numbered 2, 3 and 4 but *cannot* exchange information with the computers numbered 5, 6, 7 and 8.

There are 12 possible projects (a project is represented by a list of channels to be upgraded):

1: (2, 3), (5,6); **2:** (2, 3), (6,7); **3:** (2, 3), (7, 8); **4:** (2, 3), (5,8);

5: (2, 4), (5,6); **6:** (2, 4), (6,7); **7:** (2, 4), (7, 8); **8:** (2, 4), (5,8);

9: (3, 4), (5,6); **10:** (3, 4), (6,7); **11:** (3, 4), (7, 8); **12:** (3, 4), (5,8);

Problem C: Proper Divisor

A proper divisor of an integer N is a positive divisor of N other than N itself. Given a positive integer N, we now denote S_N to be the number of proper divisors of N. Given the numbers in a range of $[L, R]$, we can define a sequence of numbers consisting of their proper divisors S_L, S_{L+1},…, S_R.

For example, given L=5 and R=10, the corresponding proper divisor sequence is 1, 3, 1, 3, 2, 3 as:

- 5 has one proper divisor, i.e., 1;
- 6 has three proper divisors, i.e., 1, 2, and 3;
- 7 has one proper divisor, i.e., 1;
- 8 has three proper divisors, i.e., 1, 2, and 4;
- 9 has two proper divisors, i.e., 1 and 3;
- 10 has three proper divisors, i.e., 1, 2, and 5;

We now examine another property of a sequence A consisting of M integers. X is said to be the K^{th} maximum number of sequence A if and only if it satisfies all following conditions:

a) X is a number in sequence A,

b) There are K numbers in A that are not smaller than X,

c) There are M-K numbers of A that are not greater than X.

Given 3 integers L, R, and K, you are required to find an integer X in $[L, R]$ such that S_X is the K^{th} maximum number of the generated sequence S_L, S_{L+1},…, S_R. In case there is more than one number X satisfying all of the above conditions, the smallest number must be chosen.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

Each data set consists of 3 space-separated integers L, R, and K on one line ($0 < L \leq R \leq 10^7$, $0 < K \leq R–L+1$).

Output

For each data set, write in one line space-separated integers X and S_X.

Sample Input	Sample Output
2 5 10 2 5 10 3	6 3 6 3

Problem D: Gem Mining

Byteland is a famous state with precious gems. A gem field in Byteland is a square grid of M x N cells. Each cell at row x and column y, denoted by (x,y), has a raw gem. Raw gems are useless until they are composed into refined gems.

Fine Gem Agency is one of the largest gem mining companies in Byteland. This company uses super robots to accelerate the mining time as well as to improve the quality of gems. They usually drop a robot at cell (0,0) when mining a gem field. The robot will step around to mine gems. At each step, it will move from its current cell (x,y) to a new cell (i,j), where $|x-i| + |y-j| = 3$.

In order to compose a refined gem, a robot needs *two* raw gems. Because it cannot carry raw gems too far, it must compose them at their current cell or right at the next step. At every step, the robot can choose to collect the raw gem at that cell or to leave it there. However, if it decides to collect the gem, the next cell that it moves to must have a raw gem and the robot has to compose two raw gems into a refined gem at the next step.

The quality of the refined gem depends on the quality of its raw gems. The value of a refined gem is defined as the product of the values of its gem components.

The Byteland government just explored some gem fields and requested Fine Gem Agency for their proposal. You are supposed to write a program to find a mining path for each gem field, such that all raw gems will be mined and the total value of the refined gems is maximized.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

The first line of each data set contains two space-separated integers M and N ($2 < M,N < 101$) representing the size of the gem field. In the next M lines, the i^{th} line contains N space-separated integers representing the values of the gems in N cells of the i^{th} row of the mine.

Output

For each data set, write on a single line the maximum sum of the refined gems' values or -1 if there is no desirable mining path.

Sample Input	Sample Output
2 3 6 5 5 5 10 5 5	300 -1

5 5 5 5 5 5 5 5 10 5 5 5 3 3 10 10 10 10 10 10 10 10 10	

Problem E: Meeting Point

A city transportation network has n nodes numbered from 1 to n and m bi-directional routes numbered from 1 to m. The i^{th} route connects nodes u_i and v_i with w_i km in length.

Duy lives at node 1 while Tan lives at node n. It is guaranteed that there is always a path in the network between node 1 and node n. Duy and Tan are working in a joint project which they need to have a meeting everyday at one of those n nodes. Once agreed on a meeting point, they will depart for the meeting point at the same time following the respective shortest path. Whoever gets to the meeting point earlier has to wait for the other.

On a given day, depending on their choices of transportation means, you will be notified the traveling speeds of both Duy and Tan assuming that their speeds are uniform all the way. It is important to determine the meeting point everyday so as to minimize the waiting time of the one who arrives earlier.

Duy and Tan need your help in organizing their meeting for k given days (numbered from 1 to k). On the j^{th} day, Duy travels 1 km in a_j seconds while Tan travels at b_j seconds per km. Your task is to determine the smallest waiting time c_j (in seconds) of the person that arrives earlier to the meeting point on the j^{th} day.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

Each data set comes in the following format:

- The first line contains 3 integers n, m and k ($2 \le n \le 10^5$; $1 \le m \le 2\text{x}10^5$; $1 \le k \le 10^5$)
- In the next m lines, the i^{th} line contains 3 integers u_i, v_i and w_i ($1 \le u_i, v_i \le n$; $1 \le w_i \le 10^6$)
- In the next k lines, the j^{th} line contains 2 integers a_j and b_j ($1 \le a_j, b_j \le 10^6$)

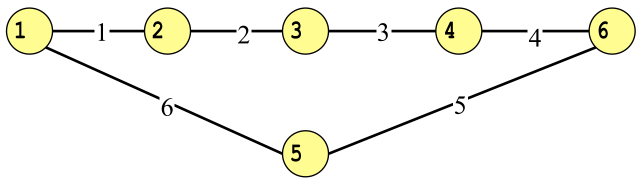
Output

For each data set, write out k integers c_1, c_2, .., c_k with each number on a separate line.

Sample Input	Sample Output
1 6 6 2 1 2 1 1 5 6 2 3 2 3 4 3 4 6 4	7 5

5 6 5 7 4 5 5	

Explanation:



Day 1: Duy and Tan meet at node 3; it takes Duy and Tan 21 and 28 seconds respectively to travel to the meeting point. Hence, Duy has to wait for 7 seconds.

Day 2: Duy and Tan meet at node 5; it takes Duy and Tan 30 and 25 seconds respectively to travel to the meeting point. Hence, Tan has to wait for 5 seconds.

Problem F: Cipher

A hacker recently invented a new encryption method. Given an original message represented as a binary string, we can pick a random substring, and insert the reversed copy of it right after the original substring. The process is repeated until we get a sufficiently long encrypted message that (hopefully) no one will be able to crack. For example, if the current message is 0*110*10, and the selected substring is 110, the encrypted message would be 0*110***011**10 (note how the reversed string is inserted into the original message).

Given an encrypted message using the above-mentioned method, you task is to determine the length of the shortest possible original message.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 100. The following lines describe the data sets.

Each data set has one string representing the encrypted message on a single line. The string consists of only digits 0 and 1 and contains at most 100 digits.

Output

For each data set, write on one line the length of the shortest possible original message.

Sample Input	Sample Output
2 011001110 0	2 1

Problem G: Object Clustering

Object search is a very challenging problem in image retrieval. To improve the speed, Alex, a computer scientist, decides to cluster images containing objects based on the structure of the objects. Currently, he has written a two steps module to detect the structure of an object in the form of a tree:

- Identify the important points in the object image.
- Connect these points together by using the least number of segments (i.e., using n-1 segments to join n points together where one segment connects two points) and still preserve the shape of the object.

The objects with similar structures are then clustered into a group. Two object structures are considered similar if and only if we can morph (change) one structure into the other by moving points around while still maintaining the connections between points in the structure.

To formalize the definition of structure similarity, let us assign n labels 1, …, n to n points of a structure. The signature of a labeled structure is the sorted sequence of n-1 segments of the structure

$$[(a_1, b_1), (a_2, b_2), \dots (a_{n-1}, b_{n-1})]$$

where

- (a_i, b_i) are segments in the structure ($1 \le i \le n$);
- $a_i < b_i$ $\forall i$: $1 \le i < n$;
- either $(a_i < a_j)$ or $(a_i = a_j)$ and $(b_i < b_j)$ $\forall i,j$: $1 \le i < j < n$;

Two structures are considered similar if and only if two following conditions are satisfied:

1. They both have the same n number of points.
2. For each of these two structures, there is a way to assign n labels 1,…, n to their points in such a way that the corresponding labeled structures have the same signature.

For example, in the picture below, there are four objects where the two left most objects are similar and the two right most objects are also similar.

Given some object structures, your task is to determine the number of similar object clusters. Specifically, an object structure is given with one way of labeling. In order to determine if two object structures are similar, you have to decide if there is an appropriate way of re-labeling these structures in order for them to have the same signature.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

The first line of each data set contains an integer K ($1 \le K \le 1000$) that is the number of objects. The next $K*N$ lines describe K objects where one object is described in N consecutive lines. The first line of each object description is the number of important points, called N ($1 \le N \le 1000$). Each of N - 1 following lines contains two integers a and b ($1 \le a,b \le N$) which are labels of a segment's end points.

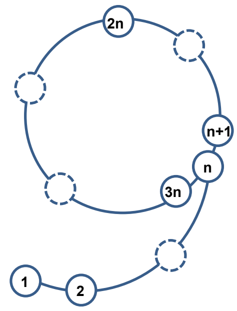
Output

For each data set, write on a single line an integer which is the number of similar object clusters.

Sample Input	Sample Output
1 3 4 1 2 2 3 3 4 4 1 3 3 2 2 4 4 1 2 1 3 1 4	2

Problem H: Welcome

To welcome this year's 96 teams to Danang, the ACM ICPC Committee organizes an opening ceremony at the city stadium. The ceremony starts with an art performance extravaganza. Joining the performance is a large groups students coming from the hosting university who will stand in the designated positions to make a 96-numbered shape. Each digit 9 and 6 has $3n$ positions numbered from 1 to $3n$ (i.e., there are $3n$ students in total for each shape). The picture on the right is the shape of the digit 9, and the shape of the digit 6 can be configured by rotating the shape of the digit 9 by 180 degree. A unique integer ranging from 1 to $3n$ is marked on each student's jacket (there are no two students carrying the same number on their jackets). On each digit configuration, the positions are arranged as followed which is also illustrated in the picture on the right:



- the student at position 1 stands ***next*** to the student at position 2
- the student at position i stands ***next*** to the students at positions (i-1) and (i+1), where $1< i < 3n$
- the student at position $3n$ stands ***next*** to the students at position ($3n$-1) and n.
- Notice that the student at position n stands ***next*** to students at positions (n-1), (n+1) and $3n$.

There will be performance actions on the configurations of digits 9 and 6; however as the configurations of the two digits 9 and 6 are symmetrical, we only illustrate the performance actions on the configuration of the digit 9.

The performance will be controlled by the commands shown on the big electronic board with one command at a time. There are two types of commands:

- The Swap command - the electronic board will display "S u v" where u and v are two integers, $u \neq v$ and $1 \leq u,v \leq 3n$ – to direct two students carrying numbers u and v to swap their positions with each other. The Swap commands are always *valid*.
- The Flower command – the electronic board will display "F p q" where p, q are two integers, $1 \leq p \leq q \leq 3n$ and $q - p < n$ – to direct the group of students (called group p-q) carrying the number x satisfying the condition $p \leq x \leq q$ to wave flowers in a certain way. The Flower command is *valid* if and only if all students in this group p-q form one connected group where two students are considered connected if they are standing ***next*** to each other.

The commands list to be displayed on the electronic board was installed on a computer controlling the electronic board. Everything is ready for the performance, but something has unexpectedly happened. The director got bird flu H5N1 and is immediately put in a strictly isolated hospital! No one knows what the initial arrangement of the students standing positions is. The organizer has to try some initial arrangements and check if all commands installed on the computer are valid when executed in order.

You are required to check if a Flower command to wave flowers is valid. Given a commands list, the output of the check will be a string S containing letters from the set $\{C, U\}$, where the i^{th} letter corresponds to the check result of the i^{th} Flower command in the list: S_i will be C if the command is valid, otherwise S_i will be U.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets. Each data set is described as follows:

- The first line contains an integer n ($2 \le n \le 10^5$);
- The second line contains $3n$ integers $a_1, a_2, \ldots, a_{3n}$ representing an initial arrangement where the student carrying number a_i stands at location i;
- The third line contains an integer m ($1 \le m \le 10^5$) representing the number of commands;
- The i^{th} line in the next m lines contains a statement of the form $Z\ k_1\ k_2$ representing the i^{th} command, where Z is a character from the set {'S', 'F'}; k_1 and k_2 are two integers.

The numbers on a line are separated by at least a space.

Output

For each data set, write in one line the resulting string S.

Sample Input	Sample Output
1 5 14 11 13 8 7 3 2 4 1 10 12 15 9 6 5 8 F 1 4 F 5 9 F 5 6 S 10 5 F 1 5 F 10 12 S 5 11 F 11 12	CCCCUC

Problem I: Geeky Fun Fact

Geeks love to analyze things, often to the point where non-geeks would call it excessive. Those of us with a mathematical bent tend to like finding interesting facts about each year as it arrives, since virtually every number has something interesting to note about it. To save you some time, here are four facts about 2013 that you can memorize and use tonight to impress the friends, family, acquaintances, and/or possibly complete strangers with whom you're planning to ring in the New Year. If it's after midnight as you read this, there's always New Year's brunch conversation, right?

1. *2013 is composed of four different digits, and is the first such year since 1987.*
2. *2013 is composed of four sequential digits, although obviously not in order. The last such year was nearly 600 years ago, back in 1432. But the next such year is only 18 years away.*
3. *2013, 2014, and 2015 are consecutive years each of which is the product of three distinct primes (3 x 11 x 61, 2 x 19 x 53, and 5 x 13 x 31, respectively). The last such three-year sequence was back in 1885-1887, and the next one isn't until 2665-2667.*
4. *As was 2012, 2013 is one of only 45 multi-digit numbers that, when spelled out in English, are alliterative (i.e., "two thousand thirteen").*

(www.wired.com)

Geeky Jerry has just read about these interesting facts but it is already November of 2013 so he has no chance to show off these facts. He realizes that the first fact is also correct with next year – the year of 2014 and he wants to know more about other years having such property.

Your task is to help Jerry write a program that given a year A, it should return the next year B - the smallest year that is greater than A - such that each of its digits is unique.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets. Each data set consists of a positive integer A ($A < 10^9$).

Output

For each data set, write in one line the number B – the next year after A where each of its digits is unique.

Sample Input	Sample Output
2 1987 2013	2013 2014

Problem J: Number Game

Let us play a number manipulation game where the players are given a sequence of distinct positive integers $a_1, a_2, \ldots, a_n$ and a positive integer k. Starting with a_1, the players need to perform at most k moves in order to obtain a_n. At every move, a_i can be changed to a_j ($i \neq j$) if ($6 \text{ x } a_i + a_j$) is a prime number.

Given a sequence $a_1, a_2, \ldots, a_n$ and two positive integers k and M, let us denote W to be the number of ways to obtain a_n from a_1 using at most k moves. Your task is to compute the remainder of W when divided by M.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not greater than 20. The following lines describe the data sets.

Each data set consists of two lines where the first line contains 3 space-separated integers n, k, M ($n \leq 20$; $k,M \leq 10^{12}$). The second line contains n space-separated positive integers $a_1, a_2, \ldots, a_n$ ($a_i \leq 10^9$).

Output

For each data set, write on one line the required remainder.

Sample Input	Sample Output
1 3 2 100 1 5 7	2