1st CSE Carnival'17 MBSTU

www.cse.mbstu.ac.bd

10, 11 February

There are in total 15 pages including this title page.

11th February, 2017

A - Moving Circles

Consider a square and two circles drawn in it. The lower left corner of the square is **(-1, -1)** and the upper right corner is **(1001, 1001)**. Both circles have same radius **R = 1**, and centers at **(x1, y1) and (x2, y2)**. They start moving from their initial positions in the direction of two vector **D1** and **D2** respectively, and with a speed **V**.

If a circle hit the boundary of the square, it will reflect according to basic reflection law, i.e. the direction before and after hitting will make equal angle with the line perpendicular to the boundary at the hit point. If the circles collide with each other in their path, they will both vanish.

A collision is detected at any moment if from that moment the circles will have common points for a duration greater than 0.01 second. Input file will contain only such collision cases. Note that, two touching circles are not considered as collided.

Initially, the circles will start moving at the given speed and according to the direction vectors from a non-intersecting static state. Given the information, you have to determine the positions of the circles after **T** seconds.

Input

The first line will contain an integer C denoting the number of test cases. The following C lines will contain 10 space separated integers denoting x1, y1, x2, y2, V, horizontal component of D1, vertical component of D1, horizontal component of D2, vertical component of D2 and T respectively.

Constraints

- 0 < C <= 5000, 0 <= V <= 100, 0 <= T <= 100
- x1, y1, x2, y2 will be such that the circles fit in the square. i.e. in range [0,1000] inclusive.
- Absolute value of components of D1 and D2 <= 100

Output

Print Case no. then 4 space separated values rounded to 4 decimal places denoting the center positions of the first circle and the second circle respectively. If they collide in their path during the given time span, print "Vanish" without quotes in a single line and print the center positions rounded to 4 decimal places at the moment they are just about to vanish. See sample for more details.

Sample Input	Output for Sample Input

2
5 5 10 5 1 0 1 -1 0 5
5 5 10 5 1 1 0 -1 0 5
	Case 1:
	5.0000 10.0000 5.0000 5.0000
	Case 2:
	Vanish
	6.5000 5.0000 8.5000 5.0000

B - Clock Hazard

Little Q was born in a digital world! She only knows how to see time in a digital clock. Recently she is learning about analog clocks. To test this new skill, her grandfather has given her two timestamps in two special clocks. The special clocks are identical circles, have no digits printed and only two hands; one smaller hand for hour and one bigger for minute. She has to determine whether the given two clock states are distinguishable or not.

Two clocks are NOT distinguishable if by rotating any one of them the two clocks look same. Note that, you can't flip the clocks. Moreover, little Q has sharp eyes but not that sharp to distinguish between angles with difference less than 0.05. So, practically two angles A and B will be considered equal if, $|A - B| < 0.05$

Input

The first line will contain an integer T, the number of test cases. Next T lines will contain 6 integers h_1, m_1, s_1, h_2, m_2 and s_2 in a specific format. See the sample for detail. Here h_i, m_i and s_i are the hour, minute and second of the i'th clock. All times will be valid and in a 24-hour time format.

Constraint

- `T <= 300000`
- All times will be given in such a way such that the difference angle between the hands will be approximately around some multiple of 10.

Output

For each test case, print the test case and "Yes" if the clock states are distinguishable, else print "No". See sample for details.

Sample Input	Output for Sample Input
3 0:16:22 0:49:6 1:21:50 14:27:17 3:00:00 9:00:00	Case 1: Yes Case 2: No Case 3: Yes

C - Terrorists in the City

You heard the name of the Motu King? He is a great conqueror. He is the king of a great country having **N** cities. There are some roads connecting the cities. The roads are of different lengths. The Motu King always stays and conducts all of his works from the capital. The capital city in the country is labeled with **1**.

Sometimes, the Motu King visits other cities to see how everything is going on. But he only visits a single city a day. He is so responsible about his country that he always moves on his foot to know about everything clearly. During this travel, he always tries to use the path that gives the shortest distance.

Days are not always too good. All on a sudden a terrorists' gang was formed to destroy the peace of the country. To irritate the King, the terrorists' gang want to destroy exactly one road in the country to make the work of the king hard. They want to choose a road in a way such that there will be no path available for any X cities where the king can travel using the shortest distance he could before. To be more precise, they want to destroy a road and as a result there will be exactly **X** such cities that the King can't visit using any shortest distance starting from the capital according to the initial graph, but the other **N-X** cities can be visited using the shortest distance.

But the gang leader is unable to choose such a road because there is no programmer in the gang. So, they came to you and offered a lot of money to help them. Though you didn't want to help the bad guys, but they threatened you to kill if you don't help them. As a result, you are bound to solve their problem and help them.

Input

The first line of the input will be the number of test cases, **T**. Each test case started with a line having two numbers, **N** and **M**. **N** is the number of cities and **M** represents the number of roads in the country. Each of the next **M** lines contains three integers describing a road, **U V W**. This means city **U** and **V** is connected together with a road of length **W**. Then there will be a line containing a single integer **Q**, representing a number of query. Each of the next **Q** lines contains exactly one Integer **X**.

Constraints

1 <= T <= 5
1 <= N, Q <= 100000
1 <= M <= min(200000, N*(N-1)/2)
1 <= U,V <= N && U != V
1 <= W <= 1000000000
0 <= X <= N-1

Output

For each test case, on the first line, you have to print the case number in the following format without quote: "Case Z:"

Here **Z** is the case number. After that you have to print **Q** lines. For each query, you have to print a single integer. The length of the road which should be destroyed such that the king can't visit exactly **X** cities using the shortest path. If there are multiple roads satisfying the condition, print the minimum length. If you can't find any edge, then just print "**-1**" without quote.

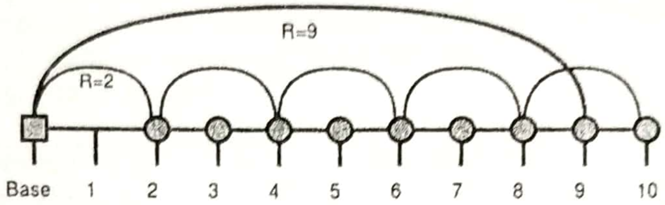
Sample Input	Output for Sample Input

1
7 6
1 2 1
2 3 1
2 4 1
4 7 1
1 5 2
5 6 1
3
2
3
4
	Case 1:
	1
	-1
	1

D - Repeater Again

Hello soldier, welcome to the military base Ground Zero. The enemies are planning to attack our base. Right now they are camping in the jungle near our camp for the night. We must destroy them before the sunrise.

Now there are N enemy vehicles in a straight line with first 1st vehicle 2 distance away, 2nd vehicle 3 distance, 3rd vehicle 4 distance and so on with Nth vehicle at N+1 distance away from the base. For all practical purpose, you can think of our base and all the vehicles are dots in a straight line.



Now the enemy doesn't know about our secret weapon the repeater. The repeater is a powerful weapon with which can destroy the enemy vehicles. When firing the repeater, we can choose a parameter R>1 and the repeater will first destroy everything at exactly R distance away, then it will destroy everything in 2R distance, then 3R, 4R until all the multiples of R less than or equal M are destroyed. Once fired, the repeater can't be stopped. Now your job is to operate the repeater.

The image is a case where the repeater is fired with R=2 and R=9 settings where N=9. Notice that it did not destroyed all the targets. We want to destroy all the enemy vehicles but we also want to minimize the number of time we need to fire the repeater.

So, soldier, your mission is to destroy all the enemy vehicles with minimum number of fire. Don't disappoint us. Note that R is an integer strictly greater than 1.

Input:

First line of the input is T(T <= 100,000), then T test cases follows. Each case have a line containing a integer N (1 <= N <= 1,000,000) the number of enemy vehicles.

Output:

For each test case print a line in "Case I: C" format where I is case number and C is the minimum number of fires for destroying all the enemy vehicles.

Sample Input	Output for Sample Input
2 1 9	Case 1: 1 Case 2: 4

E - Mathematics is Beauty – I

Have you heard the name of Carl Friedrich Gauss, the Prince of Mathematicians? He was one of the most influential mathematicians in the history of math. He had exceptional influence in many fields of mathematics and science that we have today. Gauss loved to find pattern and he was the first to find a pattern in the occurrence of prime numbers. In his 78 years of life span he worked with a lot of things. Sometimes he succeeded and sometimes he couldn't.

"Gauss gave the first clear exposition of complex numbers"

30 April 1777 – 23 February 1855

Carl Friedrich Gauss

Ashrafunnahar Eva, a famous mathematician of these days, believes that she found a pattern of such function that Gauss couldn't within his time. She wrote a code to verify her answer but it's not fast enough. Now, she wants your help. Here is the code :

```
#define MOD 1000000007
int fun(int x, int y){
    int s = 0;
    while( x > 0 && y > 0 ){
        s = (s +((x%10)*(y%10)))%MOD;
        x/=10;
        y/=10;
    }
    return s ;
}
```

```
int main(){
    int L, R, i, j, N, sum=0;
    scanf("%d %d", &L, &R);
    for(int i=L; i<=R; i++){
        for(int j=L; j<=R; j++){
            sum += fun(i,j);
            sum %= MOD;
        }
    }
    printf("%d\n",sum);
}
```

Input

The first line contains an integer, T (1 <= T <= 50,000), which denotes the number of test cases. Each of the next T lines contains two integer L and R (1 <= L <= R <= 1,000,000,000).

Output

For each test case, print the test case number and the answer modulo 1000,000,007. You must follow the same format as the sample outputs

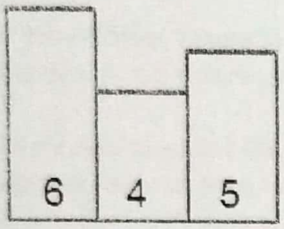
Sample Input	Output for Sample Input

3
2 5
7 10
1 1000
	Case 1: 196
	Case 2: 577
	Case 3: 60750001

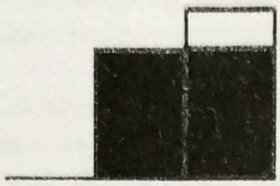
This problem is dedicated to Ashrafunnahar Eva, a brilliant student of Mawlana Bhashani Science and Technology University (Department of Mathematics) who is suffering from ovarian cancer(Stage 4). Please keep her in your prayer.

F - Pirate Dumb and Histogram

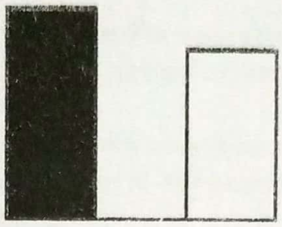
It's the night before 1st MBSTU CSE Carnival'17. Everyone is very much excited and overwhelmed. Everything is going well. Just that time, Captain Pirate Dumb broke into judges room. Being a big fan of Captain Pirate Dumb, judges want to gift him exactly one of the rectangles from the histogram that was supposed to be presented to the winner team. Now judges want to know what will be the maximum sub rectangle area if they gift ith vertical bar to Captain Pirate Dumb.



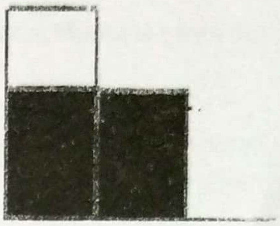
Histogram that consists of rectangles with the heights 6, 4, 5 unit where the width of the rectangles is 1.



Largest rectangle in the histogram will be 8 if judges gift 1st rectangle.



Largest rectangle in the histogram will be 6 if judges gift 2nd rectangle.



Largest rectangle in the histogram will be 8 if judges gift 3rd rectangle.

Input

Input starts with an integer T (≤ 20), denoting the number of test cases. Each case contains a line with an integer N (1 ≤ N ≤ 100,000) denoting the number of rectangles. The next line contains N space separated positive integers (≤100,000,000) denoting the heights in unit.

Output

For each case, print the case number. Then print N lines denoting largest rectangle in the histogram if judges gift ith rectangle to Captain Pirate Dumb. You must follow the same format as the sample outputs.

Sample Input	Output for Sample Input

2
3
6 4 5
3
5 2 3
	Case 1: 8 6 8
	Case 2: 4 5 5

G - Tale of Two Sequence: Majority Wins

Meera wants to become a Data Scientist. In order to learn the way of data analysis, she seeked help from the great master Data Scientist, Zinnah the Analyzer. Under the supervision of Zinnah, Meera solves various analytical problem to improve her skills. The problem Meera is solving today is called "Tale of Two Sequence : Majority Wins".

In this problem, Meera is given two arithmetic sequence by Zinnah. For each sequence, Meera is given the first term A, the difference between two consecutive values D and number of terms K.

Meera now must find the value X which occurs with highest frequency when both sequences are considered together. In case of tie, she needs to choose the smallest value for X. For example, suppose we are given $A_1 = 1$, $D_1 = 1$, $K_1 = 10$ and $A_2 = 15$, $D_2 = -3$, $K_2 = 4$, then $X = 6$.

How? $A_1 = 1$, $D_1 = 1$, $K_1 = 10$ represents the sequence [1,2,3,4,5,6,7,8,9,10] and $A_2 = 15$, $D_2 = -3$, $K_2 = 4$ represents [15,12,9,6]. If we consider both sequence together then we find that the values 9 and 6 occurs twice and all other terms (1,2,3,4,5,7,8,10,12,15) occurs only once. Since Meera needs the smallest value that occurred highest number times, she chooses 6 as X.

Zinnah believes that Meera will be able to solve the problem eventually, since she is a smart girl. Meanwhile, he decided to send the same problem to today's programming contest, for all of you to solve. Have fun :)

Input

First line of the input will be a single integer T, indicating the number of test cases. Next T lines will follow, containing 6 integers, representing A_1, D_1, K_1, A_2, D_2 and K_2. ($1 <= T, K_1, K_2 <= 10^5$, $-10^5 <= A_1, D_1, A_2, D_2 <= 10^5$)

Output

For each test case, output the test case number and a single integer value which has the highest frequency if the two sequences are merged. In case of tie, print the smallest value. See sample input/output for details.

Sample Input	Output for Sample Input

2
1 1 10 15 -3 4
1 2 10 1000 500 3
	Case 1: 6
	Case 2: 1

H - Social Media

Social media is the greatest trend of the last decade. Nowadays people spend most of their time to be social on the virtual world than the real world.

TimeKiller is one of the mostly used social media having 100,000,000 registered users. To identify each individual it provides a unique user id. Managing such a huge number of user is not an easy task. Thus to make the maintenance easier it allows to add at most **5000** people as friends for each individual. Here each relationship is bidirectional that means, if user B is present in user C's friendlist then user C will also be present in user B's friendlist.

Mr. Lekieh is one of the user of TimeKiller whose user id is **0**. Recently he has understood that TimeKiller is indeed a time killer. So, he is planning to reduce his friendlist by unfriending those friends who are comparatively unknown to him. To do so, he wants to know how many friend will be there if he unfriend those individuals from his friendlist who have less than **K** mutual friends. Mutual friends are friends the user has in common with someone else. Note that, it is possible that after removing a friend from the friendlist, the number of mutual friend of another friend may reduce and become less than **K**, and hence Mr. Lekieh will unfriend him too.

So far, TimeKiller has no such feature, that's why Mr. Lekieh requested to the authority of TimeKiller to add such feature. As all the employee of TimeKiller are currently busy on some other stuffs, the authority is seeking your help (as you are one of the most talented programmer of the world) to design the required feature.

Input

Input starts with an integer **T (<= 10)**, denoting the number of test cases. Each case starts with a blank line and two integers **N (1 <= N <= 5000)** and **K (1 <= K <= N)**, where **N** denotes number of individuals in Mr. Lekieh's friend list, **K** denotes the minimum number of mutual friends allowed by him. The next line contains **N** space separated distinct integers $U_{1,}$ $U_{2,}$ **....** U_N, where U_i **(1 <=** U_i **<= 100,000,000)**, denotes the user id of the **i**th individual of Mr. Lekieh's friendlist. Next for each individual U_i there will be two lines. The first line contains one integer F_i **(0 <=** F_i **<= 5000)**, denoting number of friends of user U_i. The second line contains **j (1 ≤ j ≤** F_i**)** space separated distinct integers $P_{1,}$ $P_{2,}$ **....** P_j, where P_j **(1 <=** P_j **<= 100,000,000)**, denotes the user id of the **j**th individual of user U_i's friendlist.

NOTE: The input for this problem is huge. Please use fast input methods.

Output

For each case of input, print a single line "Case **X**: **R**", where **X** denotes the case number and **R** denotes the size of Mr. Leikeh's friend list after removing all the individuals having less than **K** mutual friends.

Sample Input	Output for Sample Input

2
5 5
1 2 3 4 5
1
2
2
1 3
2
2 4
3
3 5 10000
2
4 10000
 	
5 1
1 2 3 4 5
1
2
2
1 3
2
2 4
2
3 10000
1
10000
	Case 1: 0
	Case 2: 4

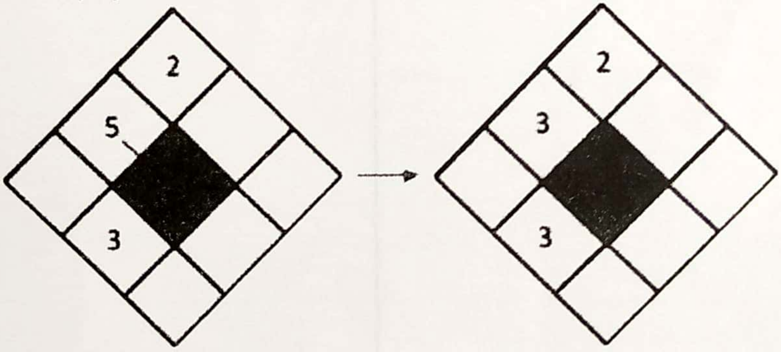
I - Game of Stones

In the land of Easteros, there has been a war going on between many houses to conquer the iron throne for years. Surely it is very daunting as many lives have been and will be lost in the process, but still some try to find peace someway or another. Dasia, the mother of kittens of house Ranheen, is a major contender of the war, who has been battling ferociously with her army of.. well kittens. Even throughout the war, she sometimes plays with her kittens just for some distraction from the battles.

Dasia currently wants to play a game with her kitten, Tutu, on a very large two dimensional diamond-shaped battlefield, which she had just conquered. The field has N x N cells. Since the whole field is diamond-shaped, so are the cells. Each of the cells might contain some rubble of stones, which were created during the fierce battle due to the mighty destructive powers of her kittens. Their powers were so CATastrophic that they created an infinitely deep crater in one of the cells, where their main threat was hiding. The game will be played in turns. On each turn, a player needs to choose a positive number of stones from exactly one of the cells and move them to an adjacent cell so that the distance between the stones and the crater is minimized. Two cells are adjacent if they share a border. Once a player moves the stones to the crater and drops them inside, the stones are lost forever. The player who can't make any valid moves on their turn, loses.

Dasia, (hopefully) being the future queen of Easteros, will always go first. But she won't play unless she knows that she's definitely going to win. For this matter she has turned to you, the royal coder, to check if it is at all possible to win with the current layout of the field. Given the layout of the field, your task is to find whether Dasia will win or not, provided that both Dasia and Tutu both play the game optimally.

Here, optimal play is such a play, which achieves the best possible result for that player.



Dasia throws 2 stones into the crater

Tutu will always lose from here

Input

The first line of the input contains a single integer T, which denotes the number of test cases. This is followed by the test cases. The first line of each test case contains two space separated integers N and M, which denotes the dimension of the diamond field and the number of cells which contain stones respectively. The next line contains two space separated integers X_h and Y_h, which denote the coordinates of the crater. The next M lines contain three space separated integers X_i, Y_i and W_i which denote the coordinates of a cell, which consists of W_i stones. The Y^{th} cell on the X^{th} row of the field has the coordinates (X, Y). **NOTE: The input for this problem is huge. Please use fast input methods.**

Constraints

- 1 <= T <= 50, 2 <= N <= 10^9, 1 <= M <= min(N * N, 10^4)
- Each W will fit in a 64-bit signed integer.
- For each X, Y pair, 1 <= X < 2 * N and 1 <= Y <= N - |N - X| (here, |a| denotes the absolute value of a)

It is guaranteed that none of the cell information for each layout will be repeated in the input, and there will be no stones in the hole initially.

Output

For each test case, output the case number, followed by either of the strings "Dasia wins! :D" or "Tutu wins -_-" without quotes if Dasia or Tutu will win the game respectively if they both play optimally. See the sample input/output for more clarification.

Sample Input	Output for Sample Input

3
3 3
3 2
4 1 3
1 1 2
2 1 5
3 5
3 2
3 1 5
4 2 6
2 2 7
3 3 8
2 1 4
4 9
3 2
5 2 55
4 4 45
1 1 21
4 2 18
7 1 11
2 2 13
6 2 41
5 3 62
4 1 27
	Case 1: Dasia wins! :D
	Case 2: Dasia wins! :D
	Case 3: Tutu wins -_-

Explanation

For the first case, if Dasia takes 2 stones from the cell (2, 1) and then throws them into the crater, the field will have two cells remaining with 3 stones each and and another one having 2 stones. After that, no matter what move Tutu makes, Dasia will always win the game. This is visualized in the image.

For the third case, no matter what move Dasia makes, Tutu will always win the game.

J - Sum of Subsequence

A sub-sequence is a sequence that can be derived from another sequence by deleting zero or more elements without changing the order of the remaining elements. For example, Let A = {3,5,5,2,6}. If B = {3,6}, C = {5,5}, D = {5,2}, and E = {6,2} then B, C and D are subsequences of A, but E is not.

An increasing subsequence is a sequence where all elements are in strictly increasing order. In the example above, B is increasing sub-sequence where C and D are not.

Now, you are given an array of N integers in non-decreasing order. You have to count the number of increasing subsequences for each length K where 1<=K<=N.

Input

First line of the input contains a positive integer N indicating the number of elements in the array. Next line contains N integers, the elements of the array.

Output

For each test print N space separated integers. As the answer can be long enough, print the answer after modulo by 1711276033. For more details, please check sample output format.

Constraints

1 <= N <= 10^5

Sample Input	Output for Sample Input
5 1 2 3 4 5	5 10 10 5 1
3 1 2 3	3 3 1
10 1 1 2 3 3 3 3 4 4 4	10 35 50 24 0 0 0 0 0 0

Explanation of second test case:
Increasing sequence of length 1: {1}, {2}, {3}
Increasing sequence of length 2: {1,2}, {1,3}, {2,3}
Increasing sequence of length 3: {1,2,3}

K - Bus Ride to MBSTU

Many of you must came from Dhaka to MBSTU by bus. Interestingly, there are many different bus services that comes to MBSTU from Dhaka. Like there is direct bus service from Mohakhali to MBSTU. There are also AC coaches from Kalyanpur to MBSTU and so on.

Some of the services have very surprising fare system. They don't have any fixed amount of fare. You must haggle with the conductor and determine what the fare will be before you start your journey. The fare can surprisingly range from 50 to 300 BDT, based on your ability to haggle.

So after getting on the bus, you were trying to determine whether you have won the haggle or not. You will win the haggle if the average fare you paid for your group is less than the average amount collected from rest of the people riding that particular bus ride. You lose otherwise.

For example, you have a group of 3 friends and you paid 300 BDT total for your ride. So on average you paid 100 BDT per person. Now there are two other groups who paid 120 BDT for 2 persons and 400 for 3 persons. So on average rest of the riders paid (120 + 400) / 5 = 104 BDT. So you guys won your haggle.

Input

First line will contain, T ($0 < T < 101$), number of test cases. Each case will start with N ($1 < N < 101$), the number of groups in the bus ride. Next N lines each will contain two integers, S ($0 < S < 101$) and X ($0 < X < 30001$). S is the group size and X is amount that group agreed to pay for the group in total. Your group is the first group in each case.

Output

Print one line for each case, “Case T: ” then followed by Win/Lose based on whether you have won your haggle. See sample for more clarification.

Sample Input	Output for Sample Input

2
3
3 300
2 120
3 400
3
3 300
2 120
3 300
	Case 1: Win
	Case 2: Lose