NDUB CSE Fest Programming Contest – 2017

A giveaway

Time Limit: 1.5 sec

Devu and his brother love each other a lot. As they are super geeks, they only like to play with …. Jeez! I'd rather give you a straightforward problem description than writing such an ugly statement. Here we go, Given N and M, find out the largest number X which satisfies the following conditions:

X = M^n where n is an integer.

And N! mod X = 0

Input:

Input starts with an integer T (T <= 25), denoting the number of test cases.
Each case starts with a line containing two integer N (1<= N <= 10 6) and M (1 <= M <= N).

Output:

For each test case, print the required number in a single line. As the number can be too large, just print the number MOD 10 7 +7(10000007). See sample output for clarification.

Sample Input:

1
5 2

Sample Output:

8

NDUB CSE Fest Programming Contest – 2017

Alien Coordinates

Time Limit: 1 sec

It is the year 4017, mankind has reached the pinnacle of science and technology, they now operate on renewable energy, cured all diseases, traveled beyond our minuscule galaxy and even came in contact with extraterrestrial sentient beings.

Bihim and you are two of lucky persons who were born in this era. One day, you decided to take a stroll through the edge of "Cosmos Redshift 7" galaxy but unfortunately, something went wrong with your ship and you crash landed into an unknown planet. There you discover a never before seen race of sentient and friendly aliens. The kind aliens offered to help you and your friend go back to earth. But to your surprise, you saw that this race of aliens uses a different coordinating system than the universal standard. How troublesome! Can you figure out the mystery of this alien coordinating system and write a program to convert it to your coordinating system?

The Problem

The coordinate system is laid out in the form of a *N*N* matrix. The universal coordinate system of size 5 is as follows:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

And the sample you got form the aliens is:

1	2	6	7	15
3	5	8	14	16
4	9	13	17	22
10	12	18	21	23
11	19	20	24	25

Find the difference between them and figure out a way to convert the alien coordinate system to universal coordinate system.

Input:

The first line of the input will be an integer *K*, the number of coordinate set you have to covert, followed by an integer *N*, denoting the size of the matrix, followed by *N*N* matrix.

$1 >= K <= 2^{32}$, $0 >= N <= 2^{16}$

Output:

Print the converted matrix one by one. Make sure to keep a blank line between the matrices.

Sample Input

2

5

1 2 6 7 15

3 5 8 14 16

4 9 13 17 22

10 12 18 21 23

11 19 20 24 25

1

1

Sample Output

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

1

NOTRE DAME UNIVERSITY
BANGLADESH

NOTRE DAME UNIVERSITY
COMPUTER CLUB

NOTRE DAME UNIVERSITY
BANGLADESH

NOTRE DAME UNIVERSITY
COMPUTER CLUB

NDUB CSE Fest Programming Contest – 2017

Atomic Counter

Time Limit: 1 sec

You were bored during your vacation, so you randomly applied for a job as a software engineer in a chemical company for giggles. But surprisingly, you got selected for the position!
On the first they of your job, they assigned you to write a software that can count the number of elements in a chemical formula. For example, the formula for Disodium EDTA is Na2C10H14N2O8. It has 2 Sodium (Na), 10 Carbon (C), 14 Hydrogen (H), 2 Nitrogen (2) and 8 Oxygen (O) atoms.

Input:

The first line of the input will be an integer *N*, the number of elements you have to process, followed by *N* lines of formula. The symbols of the elements will be two characters long at max.

Output:

For each formula, print the formula first then followed by the elements and their count one by one. Each formula set must end with a newline at the end.

Sample Input:

2

Na2C10H14N2O8

Cu(NO3)2

Sample Output:

Na2C10H14N2O8

Na : 2

C : 10

H : 14

N : 2

O : 8

Cu(NO3)2

Cu : 1

N : 2

O : 6

NOTRE DAME UNIVERSITY
BANGLADESH

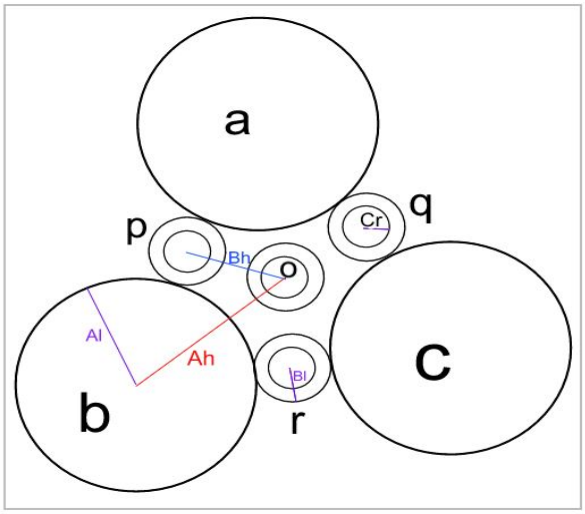
NOTRE DAME UNIVERSITY
COMPUTER CLUB

NDUB CSE Fest Programming Contest -2017

Fidget Spinner

Time Limit: 1s

Picture 1



Picture 2

Making a fidget spinner is not that hard. But for that you will need to create a 2D model so that you can create a mould afterwards. Let's begin then? You'll draw 4 identical small circles '**p**', '**q**', '**r**' and '**o**'**(outer circle)** in the paper holding the '**o**' at the center of '**p**', '**q**' **and** '**r**' where '**p**', '**q**', '**r**' are equidistant from each other and the **radius of** '**p**', '**q**', '**r**' and '**o**' is **Bl**. Then you'll draw 3 more identical circles '**a**', '**b**' and '**c**' with same **radius Al**. Each of them will touch exactly two different small circles from outer side and edge to edge (for example: '**a**' touches '**p**' and '**q**'). Then you'll draw **4 more smaller circles (inner circle inside** '**p**', '**q**', '**r**', '**o**' **)** with radius **Cr** and cut out that part as you'll put the bearings on that place later. Now, all you have to do is, cut-out the fidget model like the fidget spinner shown on picture 2. Can you find the area of your fidget spinner model?

Input: The first line of the input will be **T(1<=T<=1000)**, the number of test cases. Then there will be **T** lines each containing two real numbers **Ah, Al** with 6 digits after decimal and 3 integers **Bh, Bl, Cr** separated by single spaces in between. The input will be such that the larger circles **(** '**a**', '**b**',

‘**c**’ **)** will never coincide with circle ‘**o**’**.** You can safely assume that the input will generate a valid fidget spinner. Check out the picture 1 for better understanding.

Here,
Ah= Distance from the center of O to the center of the largest circle.
Al= Radius of the largest circle
Bh= Distance from the center of O to the center of the medium size circle(p, q, r)
Bl= Radius of circle p, q, r, o.
Cr= Radius of the smallest circle inside p, q, r, o with same center as theirs.

Where, 1<= Ah, Al, Bh, Bl, Cr **<= 10000**

Output: The output will contain **T** lines with “**Case X: Y**” (without quotation) where **X** is the number of line, starting with 1 and **Y** is an integer which denotes the **floor of the area of the fidget spinner(The shaded part of the picture 2)**.

Sample Input	Sample Output
2 512.422251 434.764219 136 25 15 917.841346 822.237598 209 11 10	Case 1: 33776 Case 2: 53469

Mr. Kris Allen and The House

Time Limit: 1 Sec

Silicon Valley is a very wonderful country. The capital of this country is San Jose. But the most amazing thing of that country is there lives a famous person named Mr. Kris Allen. He is very interested to visit all the house of that country but he has a limited money. If he wants to visit one house to another house it will take some money. Another thing is that his relatives also live there some of them are youngers and some are elders than him. So, if he visits his younger's house cost some extra money but if he visits his elder's house gets some bonus. He may start his journey at any house but he dislikes for visiting any house second time. Now your task is to determine is it possible for Mr. Kris Allen to visit all the house? If it is possible then print "YES" along with the minimum cost otherwise print "NO". But after visiting all the house if remaining money is greater than the initial money then print minimum cost "0". For more clarity see the figures below.

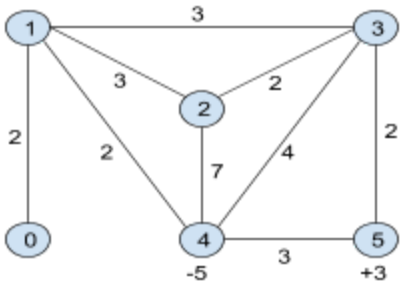


Figure: 1

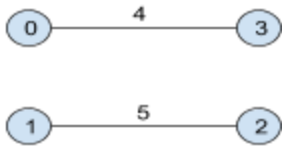


Figure: 2

In Figure 1 if his initial money is 20 then minimum cost will be 13 for visiting all the house. In Figure 2 it is not possible for him to visit all the house.

Input

The first line contains an integer **T (1 <= T <= 50)**, indicating the number of test cases. Each test case first line contains five positive integers **N, E, M, Y_h, and E_h (0 < N < 21, 0 <= E <= N, 0 <= M <= 100, 0 <= Y_h, E_h <= N)**. Indicating the number of house **(N)**, number of road **(E)**, initial money **(M)**, number of younger house **(Y_h)**, Number of Elder house **(E_h)**. Next line contains the road between two house **U, V along with cost C_i (0 <= U, V < N, 0 <= C_i <= M and U ≠ V)**. Then next line contains the younger's house **(y_i)** along with extra cost **(e_i) (0 <= y_i < N, 0 <= e_i <= M)**. Last line contains the elder's house **(E_i)** along with bonus **(b_i) (0 <= E_i < N, 0 <= b_i <= M)**. It is guaranteed that there is no house where elder and younger are live together.

Output

For each test case of input you have to print case number first, then if it is possible print "YES" without the quotation in first line and print the minimum cost in the second line. Otherwise print "NO" without the quotation. **For more clarity see the sample I/O format**.

Sample Input	Sample Output
2 6 9 20 1 1 0 1 2 1 2 3 1 4 2 2 3 2 2 4 7 3 5 2 4 5 3 1 3 2 3 4 4 4 5 5 3 4 2 12 0 0 1 2 4 0 3 5	Case #1: YES 13 Case #2: NO

NDUB CSE Fest Programming Contest – 2017

Ratul and Himalayan Guru

Time Limit: 1 sec

Ratul went to a very wise guru in Himalaya to learn about mathematics. The guru said to him, “I have a function f(x) in mind and x can be complex. I can tell you that the highest power x^{N-1} present in f(x). You can only vary x on the unit radius circle on complex plane ($x=e^{i\theta}$)”.

Ratul replied, “Then I want the value of f(x) for all values of θ where $x = e^{i\theta}$.”
The guru replied, “That my son, is not going to happen. The best I can do is to tell you the value of f(x) at equally spaced points in the complex unit circle. To say it in another way, I can give you f(x) for all $\theta = 2\pi k/N$, where $k = 0, 1, 2,\ldots, N - 1$ and $x=e^{i\theta}$.“
After thinking a while Ratul replied, “That’ll do.”
Can you do the same? Given the value of a function f(x) at equally spaced points in the complex unit circle, can you uniquely determine what the function is, where $N \leq 100000$.

Input:

If your function is called Answer_The_Guru(N,F);
Where N is the order of the function, F is the array containing the value of $f(e^{i\theta})$ for $\theta = 2\pi k/N$, where $k = 0,1,2,\ldots, N - 1$.

Output:

The output should be an array (of length N) which should contain the coefficients of the polynomials in f(x). Make sure to keep a blank line between the outputs.

Sample Input

5 101 238 2 -98 56

4 0 0 0 0

Sample Output

59.8000 + 0.0000i
53.9033 -46.3742i
-33.3033 - 2.3743i
-33.3033 + 2.3743i
53.9033 +46.3742i

0
0
0
0

NDUB CSE Fest Programming Contest – 2017

Reordering Plates

Time Limit: 1 sec

Boltu recently joined Hotel Radisson Blu Dhaka Water Garden as a part-time waiter. Every day Boltu has to do a special work, which is reordering plates as hotel Manager wants. So, every day hotel Manager gives Boltu instructions and Boltu follows that.

There are n piles of plate and m numbers of plates are randomly stationed among n piles and no piles are empty initially. Hotel Manager gives two types of instruction to Boltu:

Instruction 1 -> Change the position of plate number P from its current position to Yth position of pile number X.

Instruction 2 -> Find out the current position of plate number P

You are going to write a program which performs following instructions.

Input:

There will be T (1<=T<=50) test cases. For each test case, first you will be given two integers n, m and Q where n (1<=n<=100) is the number of piles, m (1<=m<=1000) is the total numbers of plates and Q (1<=Q<=100) is the number of queries (instructions of hotel Manager) respectively. There will be n line follows. Every line will only contain non-negative integer numbers. The first number of each line (let P) is the number of plates current pile contains. Then P number follows denoting the plate numbers from bottom to top. After that, there will be Q queries. Each query will contain a character first. If the character is 'C' then it will take three integer numbers

(P, X, Y) more as input. Means you have to change the position of plate number P from its current position to Yth position of pile number X. If the character is ‘F’ then there will be another number P which denotes the plate number. All the plate number will be distinct. No piles will be empty initially. (1<=P, Y<=m) (1<=X<=n)

Output:

For each Query of type two (instruction 2) just print two space separated integer on each line. The current pile and position of plate number P from bottom of the pile.

Sample Input:

1

2 4 4

2 1 2

2 3 4

F 4

C 2 2 2

F 2

F 4

Sample Output:

2 2

2 2

2 3

NDUB CSE Fest Programming Contest – 2017

Zig-Zag Array

Time Limit: 1 sec

Nikita has an array of integers. Each integer of the array is greater than 0 and less than 10. As you are a friend of Nikita, she wants your help for transforming the array into *Zig-Zag* array using minimum number of operations.

A *Zig-Zag* array is an array where the 1st element of the array is lesser than 2nd element of the array, 2nd element of the array is greater than 3rd element of the array, so on or vice versa.

In short, the order of the elements in the *Zig-Zag* array would be:

$[A_1 < A_2 > A_3 < A_4 > A_5 < \ldots > \ldots < A_N]$ or $[A_1 > A_2 < A_3 > A_4 < A_5 > \ldots < \ldots > A_N]$

You can apply two types of operation for transforming the array into *Zig-Zag* array.

1. You can add 1 with any one integer of the entire array.
2. You can subtract 1 from any one integer of the entire array.

You have to transform the array into *Zig-Zag* array using minimum number of operations, where all the integers of the *Zig-Zag* array will be greater than 0 and less than 10 and no two adjacent integers of the *Zig-Zag* array will be prime or composite.

Input:

The first line of the input contains an integer T (1 <= T <= 100) indicating the number of test cases. The first line of each test case contains an integer N ($3 <= N <= 10^3$) the size of the array. Next line contains N space separated integer A_i ($1 <= A_i <= 9$) the element of the array.

Output:

For each test case of input you have to print one line of output “Case #T: X”, where T is the test case number starting from 1 and X is the minimum number of operations needed for transforming the array into *Zig-Zag* array.

Sample Input:

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

Sample Output:

Case #1: 3
Case #2: 5

NOTRE DAME UNIVERSITY
BANGLADESH

NOTRE DAME UNIVERSITY
COMPUTER CLUB