ACPC
AFRICA & ARAB
Collegiate Programming
Championship

The International Collegiate Programming Contest
Sponsored by ICPC Foundation

SCPC

The 2021 Aleppo Collegiate Programming Contest

(Contest Problems)

جامعة حلب
١٩٥٨
UNIVERSITY OF ALEPPO

Syria
August 2021

Scanned with CamScanner

Problem A. The String Stretching

Input file:	`strings.in`
Output file:	`standard output`
Balloon Color:	`Gold`

Rami Husami loves the string stretching like this "margareeeeeeeeeeta".

Rami Husami gives you two strings A and B. You can stretch string B in the following way:

Choose exactly one position from B and repeat the character in this position any number of times.

For example:

if string B is "abcde" you can stretch it to become "aabcde" or "aaabcde" or "abccde"

but you can't stretch it to become "aabbcde" nor "abde".

To make Rami Husami happy you should stretch string B **(or not)** so that length of string B is maximum possible, as long as string B stays as a subsequence of A (or print -1 if you can't do this).

Note: B is subsequence from A if you can obtain B from A by removing zero or more characters from any position.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains two strings A and B. The length of each string doesn't exceed 10^5 characters.

It's guaranteed that all strings consist of lowercase English letters only and the sum of lengths of A and B over all test cases doesn't exceed 10^6.

Output

For each test case, print the maximum length of B after stretching or print -1 if you can't do this.

Example

strings.in	standard output
4 margareeeeeeeeeeta margareta aaxxbbbxxccxbbbbxc abcc blmrsadyyyyyyyy blmrsady abcd acdb	18 6 15 -1

Note

The balloon color for this problem is gold because Rami Husami is gold.

Scanned with CamScanner

Problem B. Big Cake

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

Mourad loves cakes so much (so do his friends).

Mourad has a big cake and he wants to cut it into pieces to share the cake with his friends (what a nice guy!).

He can cut the cake several times (possibly zero), and he can make each cut vertically or horizontally.

For example, if he cuts it vertically he will get 2 pieces of cake, and if he cuts it vertically and then horizontally he will get 4 pieces of cake.

Mourad has N friends (including himself), and he wants to get **at least** N pieces of cake.

Although he is a nice guy he is not so smart! so he asks you to help him to calculate the minimum number of cuts (vertically or horizontally) to get **at least** N pieces.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains a single integer N ($1 \leq N \leq 10^6$), the number of Mourad's friends.

Output

For each test case, print a single integer, the minimum number of cuts (vertically or horizontally) to get **at least** N pieces.

Example

cake.in	standard output
3 1 5 14	0 3 6

Scanned with CamScanner

Problem C. Segments And Regions

Input file:	segments.in
Output file:	standard output
Balloon Color:	Light Green

As usual in the Peaky Blinders team, Mouhanad opens Codehorses and he chooses the hardest geometry problem on it and sends it to Tony then says "Tony! solve this problem, it's easy".

This time he gave Tony the following problem, you are given N segments, the i_{th} segment is described by four integers (X_{i1}, Y_{i1}) (X_{i2}, Y_{i2}) the beginning and the ending of the i_{th} segment.

And each segment is parallel to OX or parallel to OY $(X_{i1} = X_{i2})$ or $(Y_{i1} = Y_{i2})$

Your task is to count how many regions are formed by these segments

The region should be surrounded by segments from all directions and any two points in this region can reach each other without crossing any segment.

Tony wants to make Mouhanad happy, so he asks you to count the number of regions.

Input

The first line contains a single integer T $(1 \leq T \leq 100)$, the number of test cases.

Each test case contains a single integer N $(1 \leq N \leq 10^5)$, the number of segments.

Each of the next N lines contain four integers (X_{i1}, Y_{i1}) (X_{i2}, Y_{i2}) $(0 \leq X_i, Y_i \leq 10^9)$, denotes the beginning and ending of the i_{th} segment.

It's guaranteed that each segment is parallel to OX or parallel to $(X_{i1} = X_{i2})$ or $(Y_{i1} = Y_{i2})$, and there is no intersection between any two horizontal segments or between any two vertical segments, and either $(X_{i1} \neq X_{i2})$ or $(Y_{i1} \neq Y_{i2})$, and the sum of N over all test cases doesn't exceed 2×10^5.

Output

For each test case, print a single integer, the number of regions formed by these segments.

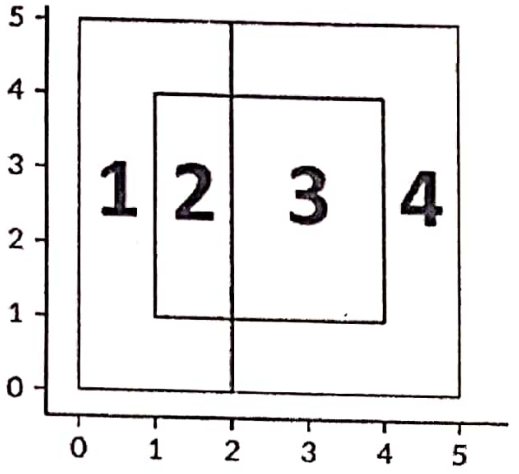
Example

segments.in	standard output
2 9 0 0 5 0 0 0 0 5 5 0 5 5 0 5 5 5 1 1 4 1 1 1 1 4 4 1 4 4 1 4 4 4 2 0 2 5 7 1 1 4 1 1 1 1 4 1 4 4 4 4 1 4 4 1 3 4 3 2 2 4 2 3 1 3 4	4 5

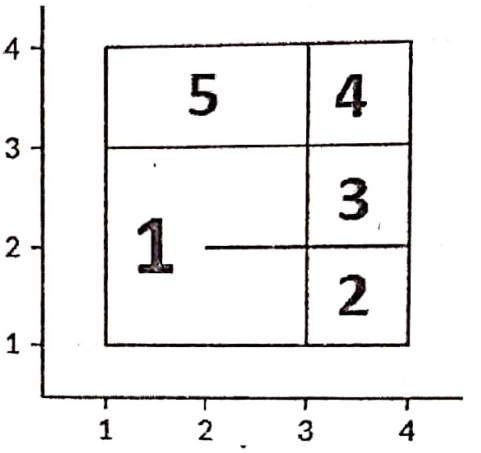
Scanned with CamScanner

Note

In the first test case, the answer is 4, and in the second one, the answer is 5



First test case



Second test case

Scanned with CamScanner

Problem D. Banana Trees

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

Moussa has N banana trees, and for each tree, he knows two integers (X_i, T_i) where X_i is the position of the i_{th} tree in OX axis and T_i is the number of minutes it takes for bananas to ripen on the i_{th} tree and be ready to be picked. (You can't pick bananas from the i_{th} tree before T_i minutes have passed).

Nawar (Moussa's friend) decided to help his friend to pick bananas from all trees.

Nawar and Moussa start from point P, and they can move to left or right (if Nawar or Moussa is in point X he can move to $X+1$ or $X-1$) and each move takes one minute.

They start picking bananas together and each of them moves independently and after they pick all the banana trees, they should return to point P.

Nawar and Moussa wonder what is the minimum time needed to pick all bananas if they start together from point P and they move in an optimal way and then return to point P.

Now, your task is to answer Q queries:

If Nawar and Moussa started from point P and they move together in an optimal way, what is the minimum time they needed to pick all banana trees?

Note: Banana picking time is very small, so it isn't taken into consideration and you can remain standing on any point for any time without moving.

Input

The first line contains a single integer TC ($1 \leq TC \leq 100$), the number of test cases.

Each test case contains a single integer N ($1 \leq N \leq 10^5$), denotes the number of banana trees

Each of the next N lines contains two integers X_i ($-10^8 \leq X_i \leq 10^8$) denote is the position of the i_{th} tree and T_i ($0 \leq T_i \leq 10^8$) denotes the number of minutes it takes for bananas to ripen on the i_{th} tree and be ready to be picked.

The next line contains a single integer Q ($1 \leq Q \leq 10^5$) denotes the number of queries.

Each of the next Q lines contains a single integer P_i ($-10^8 \leq P_i \leq 10^8$) the initial position of Nawar and Moussa.

The sum of N and Q over all test cases doesn't exceed 2×10^5

Output

For each test case, and for each query in this test case, print a single integer denotes the minimum time they needed in minutes to pick all banana trees.

Example

banana.in	standard output
1 3 1 2 2 1 3 4 2 0 3	7 4

Scanned with CamScanner

Problem E. Tony's Monsters

Input file:	monsters.in
Output file:	standard output
Balloon Color:	Red

Tony loves raising monsters in his house garden.

He has N monsters, the i_{th} monster has two values A_i, B_i (his power and his endurance)

Tony has a war with Alafandy, and he should choose a set of monsters.

The power of the set is the average power of monsters in it.

Tony should choose a valid set so that, the endurance of each monster is greater than or equal to the power of the set.

For Example: If you choose monsters $1, 2, 4$ for each B_1, B_2, B_4 will be greater than or equal $\frac{A_1+A_2+A_4}{3}$

Help Tony to choose a valid set with the maximum size possible.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains a single integer N ($1 \leq N \leq 10^5$), the number of Tony's Monsters.

Each of the next N lines contains two integers A_i ($1 \leq A_i \leq 10^9$) and B_i ($1 \leq B_i \leq 10^9$) denotes the power and endurance of the i_{th} monster.

It is guaranteed that the sum of N over all test cases doesn't exceed 2×10^5.

Output

For each test case, print a single integer, the maximum number of monsters that Tony can choose.

Example

monsters.in	standard output
1 5 10 2 8 10 5 7 7 4 8 7	3

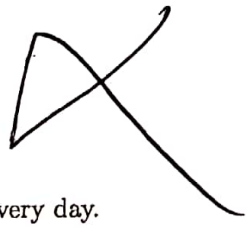
Note

In the first test case:

you should choose these monsters $2, 3, 5$

Scanned with CamScanner

Problem F. Running Race



Input file:	racing.in
Output file:	standard output
Balloon Color:	Dark Blue

Tony loves watching running races, and he watches them every day.

Yesterday Tony was watching a running race, but today he couldn't remember the final ranking.

Tony is a smart kid, for every race he watches, he writes notes about it.

In each race he writes the initial standing of the N contestants and K events, each event denotes that the contestant with the number X advances one position forward (It's guaranteed that this contestant is not in the first place and events occur asynchronously i.e. not at the same time).

Considering the initial position of the contestants and the set of events that occurred, help Tony find the final ranking.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case starts with two integers N ($1 \leq N \leq 10^5$), the number of contestants in the race, and K ($1 \leq K \leq 10^5$) the number of event.

The next line contains a **permutation** of N numbers denotes the initial standings of the contestants. the leftmost number is in the first place and the second leftmost number is in the second place... and so on.

Each of the next K lines contains a single integer X denotes that the contestant with the number X advances one position forward. (It's guaranteed that this contestant is not in the first place).

It's guaranteed the sum of N and K over all test cases doesn't exceed 10^5.

Output

For each test case, print a permutation of N numbers denoting the final ranking after the K events have occurred.

Example

racing.in	standard output
1 4 5 4 2 1 3 3 3 2 2 1	2 4 1 3

Note

In the first test case:

After the first event, the order becomes $4, 2, 3, 1$

After the second one becomes $4, 3, 2, 1$

After the third one becomes $4, 2, 3, 1$

After the fourth one becomes $2, 4, 3, 1$

After the fifth one becomes $2, 4, 1, 3$

Scanned with CamScanner

Problem G. Grid Matching

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

Marcel was teaching Memo (his little brother) string algorithms like KMP, Aho corasick, Suffix array, and Suffix automata.

After he taught him all these things, he gave him this easy problem:

He gave him a $N \times M$ grid, each cell contains a lowercase English letter, and he gives him string S which contains **distinct lowercase English letters**, and he asked him to choose a column or a row that contains S as a substring then cross out S from it.

if Memo chooses a row, he should select substring from left to right equal to S and cross it out, and if he chooses a column, he should select substring from top to bottom equal to S and cross it out.

Marcel asked Memo to calculate the maximum times he can cross out string S from the grid knowing that **he can't cross out any cell more than once**.

Memo is too young and doesn't want his brother Marcel to know about his failure.

Help Memo to calculate the maximum times he can cross out string S from the grid.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains two integers N ($1 \leq N \leq 1000$), and M ($1 \leq M \leq 1000$) the number of rows and number of columns in the grid.

The next N lines contain strings with length M consist of lowercase English letters that denotes the grid characters.

The next line contains a single integer that denotes the length of string S ($1 \leq |S| \leq 10$).

The next line contains the string S.

String S contains **distinct lowercase English letters** and its length doesn't exceed 10.

It is guaranteed that the sum of $N \times M$ over all test cases doesn't exceed 2×10^6.

Output

For each test case, print the maximum times you can cross out string S from the grid.

Scanned with CamScanner

Example

grid.in	standard output
3 3 5 cbabc babca abcaa 3 abc 4 5 aabcd bbzzz cczzz ddzzz 4 abcd 2 1 x y 2 yx	3 2 0

Note

In the first test case, the best solution is:

cb~~abe~~

b~~abe~~a

~~abe~~aa

so the answer is 3

And in the third test case,

The answer is 0 because you can only cross substrings from top to bottom.

Scanned with CamScanner

Problem H. Maximum Value Of Set

Input file:	subset.in
Output file:	standard output
Balloon Color:	Dark Green

Mouhanad loves data structure and has decided to make a test for his students.

He has given his students an array with N integers and asked them to choose a subset from these integers with a size greater than or equal to one so that the value of the set is the maximum possible.

The value of a set is:

(Sum all elements in the set) - (the maximum element in the set - the minimum element in the set)

Help Mouhanad's students to find a set with the maximum value possible.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains a single integer N ($1 \leq N \leq 10^5$), the number of elements in the array.

The next line contains N integer numbers $A_1, A_2, ..., A_N$ ($1 \leq A_i \leq 10^9$), the elements of the array.

It is guaranteed that sum of N over all test cases does not exceed 10^5

Output

For each test case, print a single integer that denotes the value of the chosen set.

Example

subset.in	standard output
2 6 1 10 2 11 3 12 1 10	31 10

Note

In the first test case:

If you choose this set $[10, 11, 12]$, the value will be $(10 + 11 + 12) - (12 - 10)$. You can't choose another set with a value greater than 31, so the answer is 31.

Scanned with CamScanner

Problem I. Huge Power

Input file:	`math.in`
Output file:	`standard output`
Balloon Color:	`Pink`

Hala loves making monsters in her house garden..!!. Each day, she makes a new one. The monster who was made on the i_{th} day has a_i *power*.

Hala noticed that each new monster has more power than all the old monsters' powers combined. After hard calculations, she concluded that all monsters give *their powers* to the new monster and Hala supports the new monster by *double of the average* of the old monsters' powers, mathematically:

$$a_n = a_{n-1} + a_{n-2} + \ldots + a_1 + 2 \times \frac{a_{n-1} + a_{n-2} + \ldots + a_1}{n-1}$$

Hala is in the middle of the fight with her Mom, so she asks you to determine the sum of the first n monsters' powers modulo $10^9 + 7$.

(We assume that the first monster has 1 power; $a_1 = 1$).

Input

The first line contains a single integer T ($1 \leq T \leq 10^5$), the number of test cases.

Each test case contains a single integer n ($1 \leq n \leq 10^{10}$).

Output

For each test case, print a single integer, the sum of the first n monsters' powers modulo $10^9 + 7$.

Example

math.in	standard output
2 2 5	4 80

Note

For the first test case: the first monster has 1 *power* the second has $1 + 2(\frac{1}{1}) = 3$, the total is 4

Scanned with CamScanner

Problem J. Trading Crypto

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

You want to trade cryptocurrencies to make some profit, you joined an exchange that sends you N opportunities, the opportunity consists of *BaseCrypto*, *QuoteCrypto*, and *Price*, you can use any opportunity in one of the two ways:

1. To buy any amount you want from the *BaseCrypto* with the $Price \times QuoteCrypto$.
2. To buy any amount you want from the *QuoteCrypto* with the $\frac{BaseCrypto}{Price}$.

Once you receive a new opportunity you can't make use of the previous one(s) anymore.

You have an amount X of *InitialCrypto* and you have N opportunities, and you know all the opportunities, write a program to determine the maximum amount of *InitialCrypto* you can get after using the opportunities in the best way.

Note: if after the last opportunity you have some cryptos other than the *InitialCrypto*, their values don't count.

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains X ($1 \leq X \leq 100$), *InitialCrypto*, and N ($1 \leq N \leq 10^5$) the number of opportunities.

Each of the next N lines contains an opportunity that consists of 3 values *BaseCrypto*, *QuoteCrypto*, and the *Price* (a real number between 10^{-4} and 10^4 and has at most 4 decimal places).

It's guaranteed that and the sum of N over all test cases doesn't exceed 2×10^5 and each of (*InitialCrypto*, *BaseCrypto*, *QuoteCrypto*) are a string that contains only uppercase English Letters and its length is exactly 3.

Output

For each test case, print a single integer, the maximum amount of the *InitialCrypto* you can get using the opportunities. Your answer is considered correct if its absolute error does not exceed 10^{-4}.

It is guaranteed that the output is smaller than 10^6.

Example

crypto.in	standard output
1 100 BTC 9 XRP ETH 0.0006 ETH BTC 0.04 ADA ETH 0.001 XRP ETH 0.001 ETH ADA 900 ADA BTC 0.0001 BTC ETH 32 AXD GTR 10 FDS YTR 20	250.0000

Scanned with CamScanner

Note

In the first test case:

You start with 100 BTC and there are 9 opportunities,

And the best scenario is:

- To buy 2500 ETH with 100 BTC in the second opportunity
- Then you buy 2500000 ADA using the 2500 ETH in the third opportunity
- Finally, we buy 250 BTC with the 2500000 ADA in the sixth opportunity.

Scanned with CamScanner

Problem K. Bad Nodes

Input file:	`graph.in`
Output file:	`standard output`
Balloon Color:	`Purple`

Coach Zain loves graph problems and solves them every day.

Today he solved this easy problem very fast and he asked his students to solve it.

Coach Zain gave his students a connected weighted graph containing N nodes and M edges, the i_{th} edge connect node U_i with node V_i and its length is W_i. And fixed K bad nodes in this graph.

Then he asked his students to process Q queries:

The i_{th} query contains two nodes (A_i, B_i) such that $(A_i \neq B_i)$

You should choose a path from node A_i to B_i so that the nearest bad node to this path is as far as possible.

For each query print the distance of the nearest bad node to the path from node A to B if you choose this path in an optimal way.

Note: The distance between a node and a path is defined as the shortest path from that node to any node in this path.

Input

The first line contains a single integer T $(1 \leq T \leq 100)$, the number of test cases.

Each test case contains four integers N $(2 \leq N \leq 10^5)$ the number of nodes, M $(N-1 \leq M \leq 10^5)$ the number of edges, K $(1 \leq K \leq N)$ the number of bad nodes and Q $(1 \leq Q \leq 10^5)$ the number of queries.

Each one of the next M lines contains three integers U_i, V_i $(1 \leq U_i, V_i \leq N)$ and $(U_i \neq V_i)$, W_i $(1 \leq W_i \leq 10^5)$ denote the i_{th} edge connect node U_i with node V_i and its length is W_i.

The next K lines contains K **distinct** integers denote the bad nodes and each integer is in the range between 1 and N.

Each one of the next Q lines contains two integers A_i, B_i $(1 \leq A_i, B_i \leq N)$ and $(A_i \neq B_i)$.

It is guaranteed that the graph is connected, doesn't contain multiple edges nor self-loops, all K bad nodes are distinct, and the sum of N, M, and Q over all test cases don't exceed 10^5.

Output

For each test case, and for each query in this test case, print a single integer denotes the distance of the nearest bad node to the path from node A to B if you choose this path in an optimal way.

Scanned with CamScanner

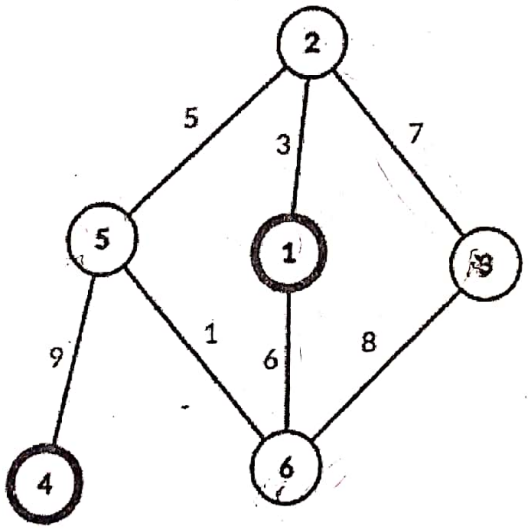
Example

graph.in	standard output
1 6 7 2 4 1 2 3 2 3 7 4 5 9 5 6 1 6 1 6 2 5 5 3 6 8 1 4 3 5 2 6 1 5 3 6	6 3 0 6

Note

In the first test case:

The bad nodes are in bold.



The answer to the first query $(3, 5)$ is 6 you should choose path $(3 \Rightarrow 6 \Rightarrow 5)$ and the nearest bad node to this path is node 1 with distance 6.

Scanned with CamScanner

Problem L. Connect Trains

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

Marcel and Naseem are bored, so they decided to play the train game.

Marcel has a train with length $L1$ and Naseem has a train with length $L2$.

They want to connect the two trains to get a long train and now they are wondering what is the length of the resulting train if they connect the two trains together?

Input

The first line contains a single integer T ($1 \leq T \leq 100$), the number of test cases.

Each test case contains two integers $L1$ ($1 \leq L1 \leq 100$) and $L2$ ($1 \leq L2 \leq 100$),

Marcel's train length and Naseem's train length, respectively.

Output

For each test case, print a single integer, the length of the resulting train if they connect the two trains together.

Example

train.in	standard output
3 23 17 15 22 1 1	40 37 2

Scanned with CamScanner

Problem M. ONE

Input file: one.in
Output file: standard output
Balloon Color: Light Blue

Naseem hates the number "one" because Naseem got "one" in his exam, but Rami Husami and Marcel love it because Mario says "ONE".

Because Marcel loves "one" he creates this problem:

You are given an integer N and you have all numbers from 1 to N

and the frequency of the i_{th} number is frq_i.

You should build an array A and each number i from 1 to N exist frq_i times in the array A, and the array A is good if and only if the difference between any two adjacent numbers in A is greater than "one"

Your task is to count how many good arrays A can be made and print answer mod $10^9 + 7$

Because Rami and Marcel love "ONE", help them to solve this problem.

Input

The first line contains a single integer T ($1 \leq T \leq 10$), the number of test cases.

Each test case contains a single integer N ($1 \leq N \leq 100$), denote that you have all numbers between 1 and N.

The next line contains the frequency array ($1 \leq frq \leq 4$), denote that the frequency of the i_{th} number is frq_i.

Output

For each test case, print a single integer, the number of good arrays mod $10^9 + 7$.

Example

one.in	standard output
2 4 2 1 1 1 6 2 1 1 2 1 1	2 408

Note

In the first test case:

Possible array A :

$[1, 3, 1, 4, 2]$

$[2, 4, 1, 3, 1]$

so the answer is 2.

Scanned with CamScanner