14’th acm ICPC Regional Contest

Tehran Site

3’th Shahed University Qualification Contest

19 Azar 1391

10 December 2012

A - Minimal K-Substring

Time Limit : 5 seconds

You are given string **S**, consisting only of lowercase English letters.

Find the lexicographically minimal k-substring of **S**.

The length of string **S** is the number of characters in it. If we denote the length of the string **S** as $|S|$, we can write the string as $S = S_1S_2...S_{|s|}$.

String $s[l... r] = S_lS_{l+1}...S_r$ $(1 \le l \le r \le |s|)$ is a substring of string $S = S_1S_2...S_{|s|}$.

We will call a non-empty substring t of string s a k-substring of s if $|t| = k$.

String $a = a_1a_2...a_k$ is lexicographically smaller than the string $b = b_1b_2...b_k$ if there exists an integer **i**($i \le k$), for each **j**($1 \le j < i$) $a_j=b_j$, and $a_i < b_i$.

Input

The first line of input contains an integer **T** (**T<101**), denoting the number of tests. There is a string **S** and an integer $k \le |s|$(separated by a space) in each of the next T lines . Length of each string is less than 1000.

Output

For each test print a single line containing the result.

Sample input

2
abcaa 3
zzbbaa 4

Sample output

abc
bbaa

B - SSC Game

Time Limit : 5 seconds

In the closing ceremony of shahed university ACM ICPC Local Contest, SSC(Student Scientific Chapter) plans to hold a game for fun .
The game is to cover a line with length L, with exactly two kinds of circles alternatively **so that the first and the last circle are same type.**
In this good and organized university there are **N** types of circles and infinite number of each one.
A line is drawn on a long table and each team must choose two kind of circles and cover the line .
The radius of circles are given in the input and no two circles have the same radius.
You have to find out how many pairs of circles can cover the line (as described above).

Input

First line of input contains an integer **T**(**T** < **101**) the number of tests .
First line of each test contains two space separated integers **L**, **N .**
There are N space separated integers in the second line denoting the radius of the circles.
(0 < L , r_i ≤ 1,000,000 , 1 < N < 101) .

Output

For each test, print a single line containing the result of that test .

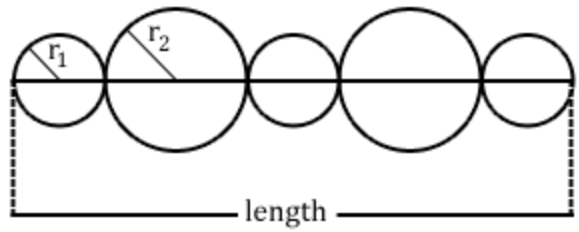
Sample input

1
20 3
1 2 3

Sample output

1

Hint : pair is (1 , 2)



C - New Flights

Time Limit : 5 seconds

In the country of programmers there are **N** cities numbered from **1** to **N**. Some of the programmers have trouble going to the other cities by train. If a programmer cannot go to another city using train, he asks the government to assign a flight from the programmer's city to his destination. You are given the list of existing trains.
Government assigns a direct flight from city **A** to city **B** if there is no path from city **A** to city **B** using existing trains. How many flights must the government assign so that all programmers become happy?
No train goes from a city to itself and no two trains have same end-points .

Input

First line contains an integer **T** (**T < 101**) , number of tests .
First line of each test contains two space separated integers **N** and **M** , the number of cities and number of trains in the country respectively .
M lines follow, each line contains two space separated integers **u** and **v** meaning that there is a train between city **u** and city **v** and another train from city **v** to city **u**.
(N < 101 , M < N*(N-1)/2)

Output

For each test, print a single number on a single line(number of flights).

Sample input

```
1
4 2
1 2
3 4
```

Sample output

```
8
```

D - Little frog

Time Limit : 5 seconds

Reza_H has a little frog which can only jump **2**, **3**, **5** or **7** units. His Frog's name is Fg.
Fg has **2** houses which are far from each other, the distance between them is **L**.
Your task is to determine the number of different paths that Fg can go from his first house to his other house.
Each path consists of a sequence of jumps, for example $P = P_1P_2 \dots P_{|P|}$.
Two paths **P** and **Q** are considered different if **|P|** and **|Q|** are different or there exist an integer **i** ($1 \le i \le |P|$) which P_i and Q_i are different**.**

Input

First line of input contains an integer **T**(**T** < **100**) number of test cases.
Each test contains a number **L**(**1 < L** < **1,000,000**).

Output

For each test, print a single line containing the number of different paths, you should print it modulo **1,000,000,007**.

Sample input

1
5

Sample output

3

E - Shahed University

Time Limit : 5 seconds

Shahed University has **N** students. Each student has many friends.
Yesterday MeHdi told one of the problems to **X**, one of the students of the university.
This student stays t_x seconds and tells the problem to all of his friends and his friends act the same.
What is the minimum time required so that all of the students know the problem?

Input

The first line of input contains an integer **T**(**T < 100**), number of tests.
First line of each test contains an integer **N,** the number of students.
There are N space separated integers in the second line, t_0 , t_1 , … , t_{n-1}.
In the third line of each test there is an integer **M** denoting the number of friendships in the university, and after that there are **M** pairs of integers **(a, b)** showing that **a and b** are friend.
In the fourth line of each test there is a integer **X** , as described above .
Students numbered from **0** to **N-1** and **M ≤ N*(N-1)/2**.

Output

For each test case print the minimum possible time that all of the students will know the problem, or print **"impossible"** instead.

Sample Input

```
2
3
1 1 1
3
0 1
0 2
1 2
1
2
17 13
0
0
```

Sample output

```
1
impossible
```

F - Garden

Time limit : 5 seconds

Reza_H's brother has a garden that its shape is like a convex polygon.
Yesterday a big stone fell on the ground from the sky.
Reza_H's brother knows the coordinates of the center of the big stone and the coordinates of the points which form the garden.
He wants to know whether the center of the stone is in his garden or not ?
Points on border of the garden are considered in the garden .

Input

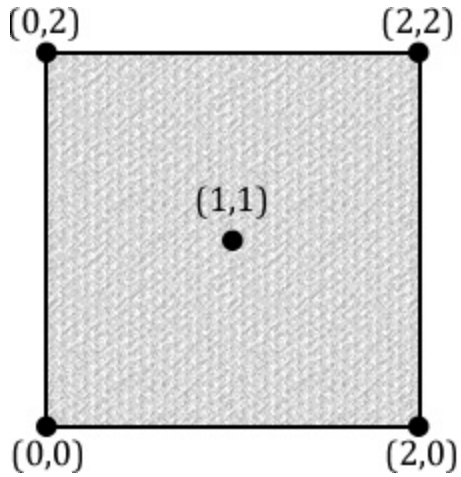
The first line contains **T** (**T < 100**) number of tests .
First line of each test contains **N** , number of points .
N lines follow , each line contains coordinates of each point .
In the last line of each test, there are coordinates of the center of the big stone .
It is guaranteed that coordinates are integer and less than **1,000,001** and **N < 101 .**

Output

For each test print a line containing "**YES**" or "**NO**" (without quotes).

Sample input

```
2
4
0 0
2 0
0 2
2 2
1 1
4
0 0
2 0
0 2
2 2
3 5
```



Sample output

```
YES
NO
```

G - Dictionary

Time Limit : 10 seconds

Reza_H has a big dictionary.
One day MeHdi and Reza_H invented a new game .
In each round of the game MeHdi writes a word on a piece of paper and Reza_H tells “YES” if a permutation of letters of the word exists in the dictionary, otherwise “NO”(without quotes).
String **A** is a permutation of string **S** if it is possible to rearrange characters in string **A** to construct string **S**.
Since you are a good programmer, Reza_H needs you to help him solve the problem.

Input

The first line contains a number **N**(**N < 100,001**) which represents the number of words in Reza_H’s dictionary .
N lines follow, each contains a word (a string up to **20** lowercase english letter).
The next line contains number **Q**(**Q < 100,001**) which represents the number of game’s round . **Q** lines follow, each line contains a query word(MeHdi’s word to be checked)

Output

For each query word print a line containing “**YES**” or “**NO**” .

Sample input

```
3
reza
mehdi
ali
4
ial
mehdiii
rezah
azer
```

Sample output

```
YES
NO
NO
YES
```

H - A New Function

Time Limit : 5 seconds

The function **f(a_1, a_2, …, a_n)**, is defined as follow

f(a_1, a_2, …, a_n) = **(a_1! * a_2! * … * a_n!)** mod 1,000,000,007

Given the values of **a_i's,** your job is to find the value of **f**

a! means that **a** * (**a-1**) * … *1

a mod b is a remainder of dividing number **a** by number **b**.

Input

First line of input contains an integer **T**(**T < 50**) number of tests .

T lines follow , each line containing an integer **n** and **n** space separated integers.

(**n, a_i < 1,000,001**)

Output

A single line for each test containing the value of **f(a_1, a_2, …, a_n)**

Sample input

```
2
2
2 3
6
27433 19903 27882 4566 22078 49
```

Sample output

```
12
268288929
```

I - Stones

Time Limit : 3 seconds

There are **N** piles of stones.
Initially each pile has $\mathbf{pile_i}$ stones .
To solve this problem you must parse two kind of queries :

Query one : 1 X Y -> Change : change the number of stones in pile **X** to **Y**.
Query two : 2 X Y -> Print : print the number of piles between $\mathbf{pile_x}$, $\mathbf{pile_{x+1}}$, … , $\mathbf{pile_y}$ which has prime number of stones.
A prime number is a number which has exactly two distinct divisors: one and itself. For example, numbers 2, 7, 3 are prime, and 1, 6, 4 are not.

Input

The first line contains **N < 100001**, number of piles .
The second line contains **N** space separated integers $\mathbf{P_i < 100001}$.
The third line contains **Q < 100001**, number of queries to process .
Q lines follow, each line contains three space separated integer **id**, **x**, **y**(**x , y < 100001**).
In all **print** queries , It is guaranteed that **x <= y** .

Output

For every **print** query print the number of piles between $\mathbf{pile_x}$, $\mathbf{pile_{x+1}}$, … , $\mathbf{pile_y}$ which has prime number of stones.

Sample input

```
3
3 3 3
4
2 1 3
1 1 4
2 1 3
1 3 13
```

Output

```
3
2
```