



Problem A – AbulHassan’s Quest

Problem Setter: M. Mohseni

“The team with the most solved problems will win the contest.”

Mr. J. Khiabani

AbulHassan is my friend. He has taken introduction to programming course this semester. He always asks me to help him with his programming assignments. Most of the times I unwillingly open his messages. Because, I don’t have much time to do his homework for him.

Last night, He texted me about his new homework question. Actually, this time I didn’t have to worry about his homework. You have to worry! :D . The Question wasn’t complicated. Here it is:

Given three integers a, b, r. Is there any integer q satisfying the following equation:

$$a = bq + r$$

Input

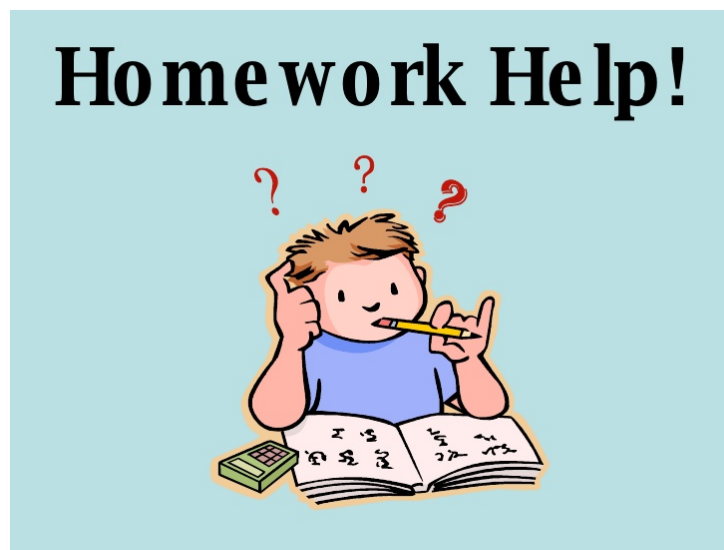
The first line of the input contains T the number of the test cases. ($T \leq 100$)

Each test case contains three integers in a single line a, b, r respectively. ($0 \leq a, b, r \leq 100$)

Output

For each test case, print “Yes” if there exists an integer q satisfying the equation. Otherwise print “No”.

Sample Input	Sample Output
3	Yes
7 2 1	Yes
25 5 5	No
7 2 2	





Problem B – Farshad’s Research

Problem Setter: M. Hassanpour

Farshad wants to do research on friendship relationships between students of the Shahid Beheshti University, so he has decided to model these relations with a graph. In this graph students are numbered from 1 to n , every student is represented by a single node, and an edge between node i and node j means that student number i and student number j are friends with each other. If i is friend with j , then j is also friend with i , and of course no student can be friend with himself. Farshad is interested in doing research on friendship groups (two students i and j are in a friendship group if and only if i and j are directly friends with each other or can indirectly connect to each other through friendship links). Farshad wants to count the number of friendship groups and the number of members each group has. Then HP comes and asks him q questions about the number of groups that exist in a particular size range. Each question is of the form (l, r) , which means HP wants to know the number of all groups that have size greater or equal to l and less than or equal to r . Help Farshad answer these q questions.

Input

The first line of input contains three space separated integers n, m, q where n is the number of students, m is the number of friendship relations ($1 \leq n, m \leq 10^5$) and q is the number of questions HP will ask ($1 \leq q \leq 10^5$). Then the next m lines each contain two space separated integers u, v . which means student number u and student number v are friends with each other. Then q lines follow, each containing two space separated integers l and r ($1 \leq l \leq r \leq n$) representing the questions HP will ask.

Output

The output should consist of q lines where line number i contains the answer to the i 'th question.

Sample Input	Sample Output
<pre> 1 6 4 2 1 2 3 4 4 5 3 5 2 4 3 3 </pre>	<pre> 2 1 </pre>



Problem C – Shortest but Hardest

Problem Setter: E. AliMohammadi

Let's assume some definitions:

1. Subarray: a non-empty continuous part of an array, eg. [1], [2], [3], [1, 2], [2, 3], [1, 2, 3] are sub-arrays of array [1, 2, 3].
2. Geometric mean: for a set of numbers x_1, x_2, \dots, x_n , the geometric mean is defined as

$$\sqrt[n]{x_1 x_2 \cdots x_n}$$

Array A consists of n non-negative integers. A is given, you have to find a subarray with the minimum value of geometric mean.

Input

The input consists one or more datasets. Integer t at the first line indicates the number of datasets. ($t \leq 100$)
In each dataset, there is one integer n that shows the length of array A , following n integers will show the elements of A . ($n \leq 100$, $0 \leq A_i \leq 10000$)

Output

For each dataset, print only one number: the answer to the problem. Print the number with exactly two digits of precision.

Sample Input	Sample Output
<pre>3 5 1 1 1 1 1 3 0 1 2 1 1000</pre>	<pre>1.00 0.00 1000.00</pre>



Problem D – Divisible Strings

Problem Setter: M. Mohseni

Mathematicians have always loved generalizing mathematics to everything. They even contributed lots of optimizations and valuable formulas to Computer Science. Have you heard about String Multiplication? What do you think will happen if we write the following code in python?

```
print (3 * "abc")
```

As you might have guessed, it prints "abcabcabc". It is equal to `print("abc" + "abc" + "abc")`

We define string S is divisible by string T , if there is some number $k \in \mathbb{N} \cup \{0\}$ which satisfies the equation $S = k * T$.

Your task is simple. Given two strings S and T . What is the minimum number of characters which should be removed from S , so S is divisible by T ?

Input

The first line of the input contains Q the number of the test cases. ($Q \leq 100$)

Each test case consists of two lines.

The first line contains string S consisting of lowercase English letters. ($0 \leq |S| \leq 10^4$)

The second line contains string T consisting of lowercase English letters. ($0 < |T| \leq 10^4$)

Output

For each test case print a single integer, the minimum number of characters which should be removed.

Sample Input	Sample Output
5	3
babbaba	7
ab	0
dictate	1
acid	14
abc	
p	
q	
dddabcbcbcacccbcbad	
abc	



Problem E – Tennis

Problem Setter: M. Nasirifar

“Who do you like more, Colonel or Morteza?” She asks.
“Colonel”. everybody, but Erfan, replies. (yes, even Morteza)

Erfan is still faithful to Morteza so he decides to be absolutely sure about this matter of life or death. He knows that there’s an upcoming tennis match between Colonel and Morteza. He will therefore choose between them based on the result of the match.

Due to the importance of the match, Colonel and Morteza are playing behind closed doors and Erfan is only be able to hear them from behind the walls. The sequence of the sounds that Erfan hears consist of the following:

- G: Ball hits the ground
- R: Ball hits a racket
- N: Ball hits the net
- ?: Unknown. can be G, R, or N

Erfan has written down the sequence of the sounds that he has heard. He wants to know who wins more games among all possible valid game sequences.

He has seen them play before so he also knows how the game is played. He knows that a **valid sequence** consists of multiple consecutive **valid scores** and the player with the **most scores wins a sequence**. (Morteza wins the sequence in case of a tie in scores)

In a valid score:

- Always Colonel starts by serving (hits the ball).
- ball hitting the net once, or the ground twice in a row ends the score. In the latter case the last player who has just hit the ball scores. It’s the other way around in the former case. This can happen after the serves and after regular hits. (That means in case of N, the ball hits the net and simply falls back in the hitter’s side, causing him to lose a point)
- a player can hit the ball if it has not yet hit the ground, or if it has hit the ground once. In case of answering a serve, the player can’t hit the ball in the air (this will be an invalid sequence) and will have to let the ball hit the ground once.
- The ball will never hit the net right after it has hit the ground.

Oh and In case of a tie in the number of possible game sequences won, Dog ate... Let’s say Morteza wins again. It is guaranteed that at least one valid sequence exists.

Input

The first line of input contains T, the number of test cases. ($T \leq 50$)

Each of the following T lines contains a single string consisting of “G”, “R”, “N”, and “?”.

length of each test case ≤ 11

Output

For each test case If Morteza is the winner, output “Morteza to doost dashtani hasti”. Otherwise output “Nasirifar to doost dashtani hasti”.

Sample Input	Sample Output
2 ??RG?? ???	Morteza to doost dashtani hasti Nasirifar to doost dashtani hasti

Explanation

Here are all the valid sequences for the first test case (scores are separated by space for clarity):

RGRGRGG
Colonel 1 : 0 Morteza

RNRGRGG
Colonel 0 : 2 Morteza

RGRGRRN
Colonel 1 : 0 Morteza

RN RGRRN
Colonel 0 : 2 Morteza

RGRGG RN
Colonel 0 : 2 Morteza

RN RGG RN
Colonel 1 : 2 Morteza
Morteza Wins in 4 of the possible valid sequences, Colonel wins in 2.



Problem F – Profiling

Problem Setter: M. Mohseni

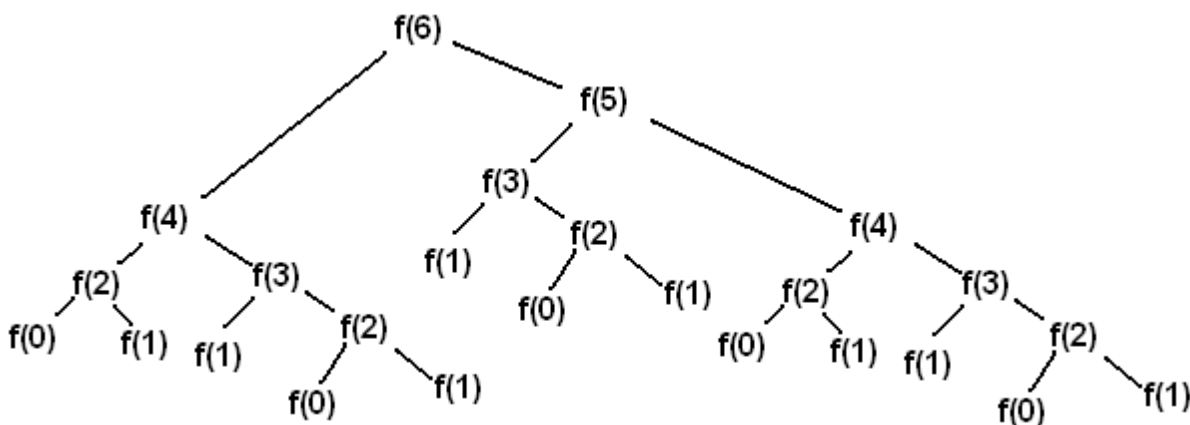
Profiling: In software engineering, profiling ("program profiling", "software profiling") is a form of dynamic program analysis that measures, for example, the space (memory) or time complexity of a program, the usage of particular instructions, or the frequency and duration of function calls. Most commonly, profiling information serves to aid program optimization.

Wikipedia

Arthur and Jacob are newly introduced to the programming world and they are trying hard for Newbies contest. Yesterday, they learned about recursive functions and Fibonacci sequence. They tried to implement the function themselves so they wrote the following code:

```
int fibonacci (int N)
{
    if(N < 2)
        return N;
    return fibonacci(N - 1) + fibonacci(N - 2);
}
```

But this program works slowly when N is a large number. They traced the program and found the cause of this problem. Take a look at the following picture:



If you want to calculate Fibonacci(6), Fibonacci(3) will be calculated multiple times!

They want to know how serious this problem can be, so they need a profiler to calculate such a thing for them.

Your task is to provide the profiler which receives two integers N, K and tells them if they call Fibonacci(N) how many times Fibonacci(K) will be calculated (according to their code).

Input

The first line of input indicates the number of test cases (There will be at most 100 test cases)

Shahid Beheshti University
Newbies Contest 2018

For each test case, there are two space-separated integers N, K ($0 \leq N, K \leq 10^5$) in a single line.

Output

For each test case, print the number of times $\text{Fibonacci}(K)$ will be calculated, if we call $\text{Fibonacci}(N)$. Since the result can be very large, print the result modulo $(\text{Mod } \%) 10^9 + 7$.

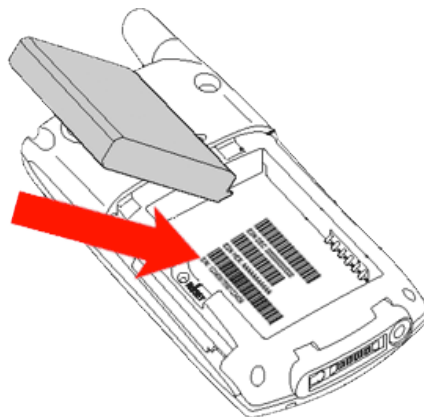
Sample Input	Sample Output
5	1
6 6	3
6 3	5
6 2	855252772
100000 3	0
5 10	



Problem G - IMEI

Problem Setter: M. Mohseni

The **International Mobile Equipment Identity** or **IMEI** /ar'mi:/ is a number, usually unique, to identify 3GPP and iDEN mobile phones, as well as some satellite phones. It is usually found printed inside the battery compartment of the phone, but can also be displayed on-screen on most phones by entering `*#06#` on the dialpad, or alongside other system information in the settings menu on smartphone operating systems.



The IMEI number is used by a GSM network to identify valid devices and therefore can be used for stopping a stolen phone from accessing that network. For example, if a mobile phone is stolen, the owner can call their network provider and instruct them to blacklist the phone using its IMEI number. This renders the phone useless on that network and sometimes other networks too, whether or not the phone's subscriber identity module (SIM) is changed.

The IMEI is only used for identifying the device and has no permanent or semi-permanent relation to the subscriber. Instead, the subscriber is identified by transmission of an International mobile subscriber identity (IMSI) number, which is stored on a SIM card that can in theory be transferred to any handset. However, many network and security features are enabled by knowing the current device being used by a subscriber.

The IMEI (15 decimal digits: 14 digits plus a check digit) includes information on the origin, model, and serial number of the device. The model and origin comprise the initial 8-digit portion of the IMEI, known as the Type Allocation Code (TAC). The remainder of the IMEI is manufacturer-defined, with a Luhn check digit at the end.

	AA	-	BB	BB	BB	-	CC	CC	CC	D
New IMEI	TAC			Serial Number			Luhn Checksum			

New style IMEI code 49-015420-323751 has an 8-digit TAC of 49-015420.

The last number of the IMEI is a check digit calculated using the Luhn algorithm, as defined in the IMEI Allocation and Approval Guidelines:

- The Check Digit shall be calculated according to Luhn formula. The Check Digit is a function of all other digits in the IMEI.

Shahid Beheshti University
Newbies Contest 2018

- The purpose of the Check Digit is to help guard against the possibility of incorrect entries to the CEIR and EIR equipment.
- The presentation of the Check Digit both electronically and in printed form on the label and packaging is very important. Logistics (using bar-code reader) and EIR/CEIR administration cannot use the Check Digit unless it is printed outside of the packaging, and on the ME IMEI/Type Accreditation label.
- The check digit is not transmitted over the radio interface, nor is it stored in the EIR database at any point. Therefore, all references to the last three or six digits of an IMEI refer to the actual IMEI number, to which the check digit does not belong.

The check digit (Checksum or Luhn Checksum) is validated in three steps:

1. Starting from the right, double every other digit (e.g., 7 → 14).
2. Sum the digits (e.g., 14 → 1 + 4).
3. Check if the sum is divisible by 10.

Conversely, one can calculate the IMEI by choosing the check digit that would give a sum divisible by 10. For the example IMEI 49015420323751?,

IMEI	4	9	0	1	5	4	2	0	3	2	3	7	5	1	X
Double every other	4	18	0	2	5	8	2	0	3	4	3	14	5	2	X
Sum digits	$4 + (1 + 8) + 0 + 2 + 5 + 8 + 2 + 0 + 3 + 4 + 3 + (1 + 4) + 5 + 2 + x = 52 + x$														

To make the sum divisible by 10, we set $x = 8$, so the complete IMEI become 490154203237518. Your task is simple, given a 14 digit IMEI, calculate its TAC and its check digit.

Input

The first line of input indicates the number of test cases (There will be at most 20 test cases)

For each test case, there is a 14-digit IMEI in a single line.

Output

For each test case, print TAC and Checksum of given IMEI respectively in a single line.

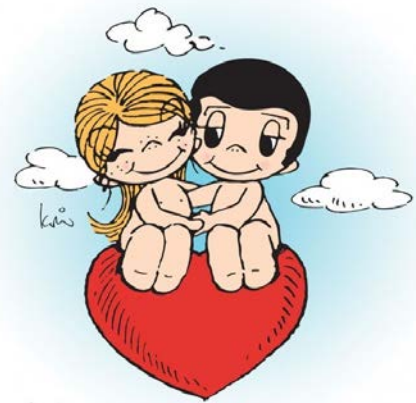
Sample Input	Sample Output
2 49015420323751 01234567890123	49015420 8 01234567 7



Problem H – Fall in Love!

Problem Setter: S. Khamesian

After a year since Erfan came back from US, he noticed that he left his heart there!!! Yes, Erfan finally fell in love. He saw Sara for the first time in Chicago last year. Now Erfan has decided to express his love and lets Sara know about his love for her (He read some psychological books and after that he made this decision). Since Sara is an intelligent girl, she has made a condition for Erfan. She asks Erfan to answer her questions. The problem is that whether $n!$ is divisible by m or not. Help Erfan to answer these questions.



Input

In the First line, one integer is given - T (the number of Sara's questions) that $0 < T < 500$. Each of the following T lines containing two positive integers, n and m , both less than 2^{31} .

Output

For each question, output a line stating whether $n!$ is divisible by m or not, in format shown below.

Sample Input	Sample Output
2 6 9 6 27	9 divides 6! 27 does not divide 6!



Problem I – Lightning Strike

Problem Setter: M. Nasirifar

There is an ancient saying that problem I is always one of the hardest problems in programming contests.

Colonel believes the above ancient saying, but what are the probabilities that you're writing the last problem of the problem set, and there is only one problem slot left, and you contact the contest coordinator to see if they can place your problem on slot I, and they respond with "I is the only remaining slot"?

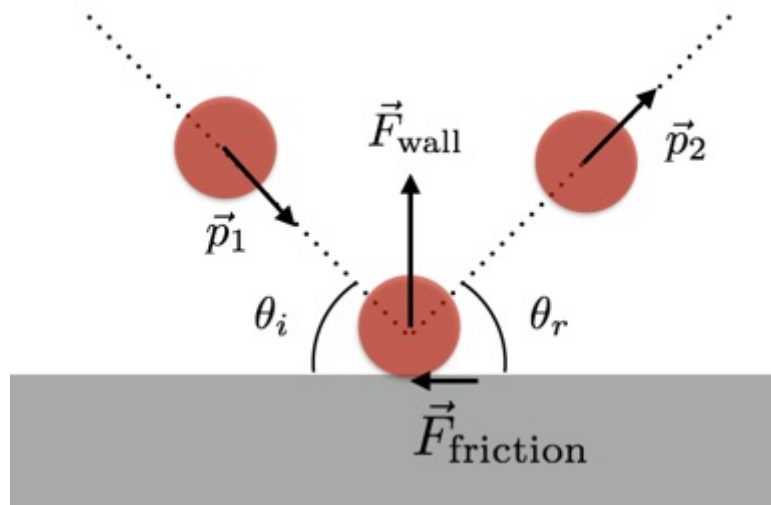
Lightning strike. Lightning strike indeed.

BUT THIS IS LAST YEAR'S STORY. This time Colonel put the dibs on writing problem I from day 1.

Colonel and Erfan were walking in TFB2DW (The Frictionless Bisected 2 Dimensional World), talking about this year's hardest problem, each of them holding a 2d tennis ball. They suddenly thought whether or not the balls will collide at some point if they were to throw them. What makes the question a little harder though, is that there is an infinite wall in the TFB2DW (hence the name). If any of the 2d balls happen to touch the infinite wall, they will behave in the normal way that 2d balls behave when touching infinite walls. (Really. What else did you expect?)

And you know what? We're asking you to give them the answer. What a surprise.

And here's the "science" if you're into that kind of stuff. There is no friction in our case, so θ_i will be equal to θ_r .



Input

There are multiple test cases in the input. The first line contains T, the number of test cases. T test cases will then follow, Each of them contains real numbers defining the starting point of each of the balls, it's moving direction, radius, and velocity(units/second), and two points which uniquely define the infinite wall. Here is the placeholder for one of the test cases in the input:

```
ball1StartX ball1StartY  
ball1DirectionX ball1DirectionY  
ball1Velocity ball1Radius  
ball2StartX ball2StartY  
ball2DirectionX ball2DirectionY  
ball2Velocity ball2Radius  
wallPoint1X wallPoint1Y  
wallPoint2X wallPoint2Y
```

All of the numbers in the input are below $1e6$. Also it is guaranteed that balls won't collide anytime after the 10^{12} th second, won't share any point with the infinite wall or each other in the initial position, and won't be immobile.

Output

For each test case, if the two balls collide at some point, output a line containing "Lightning strike", otherwise output "Not even a spark".

Sample Input	Sample Output
2 -20 4 1 -1 1 1 10 4 -1 -1 10 1 0 0 10 0 -20 4 1 -1 10 1 10 4 -1 -1 10 1 0 0 10 0	Not even a spark Lightning strike

Explanation

In the first case, the second ball having a speed equal to ten times of that of the first ball will leave the scene in a blink of an eye. In the same setup but with equal velocities (second test case), the two balls will collide after hitting the wall.



Problem J – Jitter Minimization

Problem Setter: M. Mohseni

One of the services that a reliable network provides is Jitter Minimization. Let me explain some more.

In network communication, one node is sender and the other one is receiver. Also data is carried in units called packets. So, sender sends (data) packets to receiver.

Jitter Minimization: This service guarantees that the amount of time between the transmission of two successive packets at the sender is equal to the amount of time between their receipt at the destination.

Karen is a network administrator at Computer Science and Engineering Faculty in SBU. She wants to verify whether the network is reliable enough for Newbies2018 or not. So she sent N packets from one node to another one, and wrote down the transmission and receive time for each packet on a single piece of paper. But, she forgot to write whether each entry was transmission time or receive time!!

Time to help, you are given a list of transmission and receive time of N packets. Calculate how long did it take for each packet to reach the receiver (from sender), or state that the network isn't reliable.

Input

The input file contains multiple test cases. (Number of tests doesn't exceed 100)

For each test case, first line contains a single integer N – The number of packets. ($1 \leq N \leq 600$)

The second line consists of $2N$ distinct integers t_i – the entries on Karen's paper ($0 \leq t_i \leq 10^9$)

The input terminates with the $N = 0$, this case should not be processed.

Output

For each test case print the minimum possible time between send and receive of packets, or print "Unreliable Network" in a single line.

Sample Input	Sample Output
<pre>3 1 3 5 6 4 7 2 1 3 4 6 3 1 2 3 4 12 15 0</pre>	<pre>2 2 Unreliable Network</pre>

Explanation

In the first test case:

Packet 1 is sent at $t = 1$, and is received at $t = 3 \rightarrow$ Time between send and receive = 2

Packet 2 is sent at $t = 4$, and is received at $t = 6 \rightarrow$ Time between send and receive = 2

Packet 3 is sent at $t = 5$, and is received at $t = 7 \rightarrow$ Time between send and receive = 2

All transmission times are equal to 2, so network is reliable.



Problem K – Traffic Lights

Problem Setter: M. Hassanpour

After human-beings arrived at Mars, they discovered the existence of aliens! What is more fascinating is that aliens had traffic lights just like we do. But traffic lights on Mars didn't work the way they do on the Earth. On the Earth, traffic lights follow a particular sequence, *green yellow, red, green, yellow red* and so on. Humans got bored on space and decided to play a game. They recorded the sequence of colours on a traffic light from time 1 to time n (inclusive), then created a sequence S of length N (G stands for Green, Y stands for Yellow, and R stands for Red). The value of S_i determines the colour of the traffic light at time i. Now humans want to extract a subsequence out of this sequence which should be valid according to traffic light rules on planet Earth (following the valid pattern), and have the maximum length possible. Help them achieve this task (green can be followed by green or yellow, yellow can be followed by yellow or red, and red can be followed by red or green).

Input

The first line consists of T, the number of test cases. ($T \leq 100$)

First line of each test contains an Integer N. ($1 \leq N \leq 10^5$)

Second line of each test case contains string S consisting of letters "R", "Y", "G".

Output

The output consists of T lines, each line containing the length of longest valid subsequence.

Sample Input	Sample Output
2 7 YYRGRGG 7 RRRRGYY	6 7

Explanation

In the first test case the resulting subsequence would be: "YYRRGG".





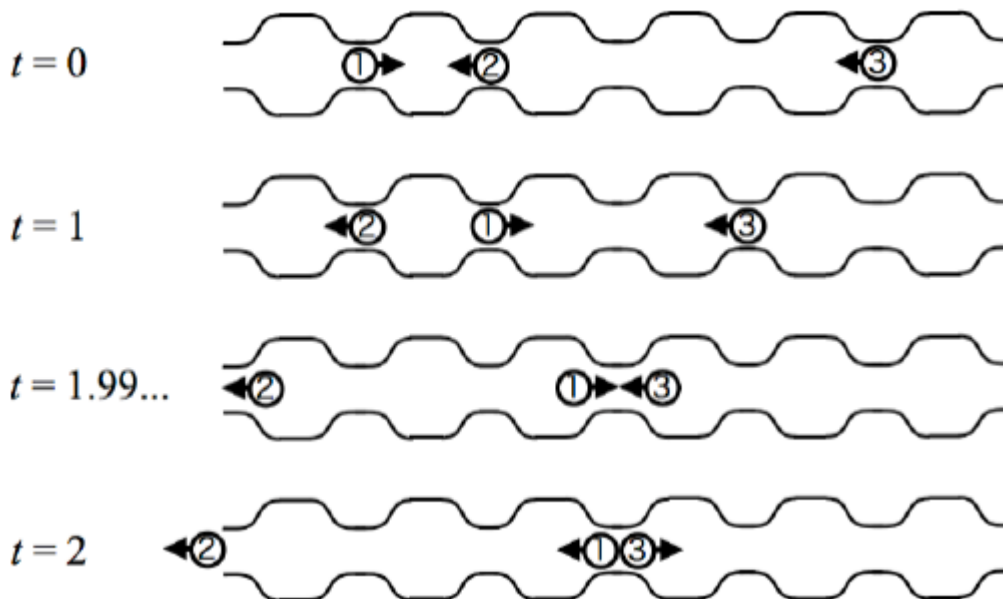
Problem L – Your House Have Ants?

(Shoma Khoonatoon Moorche Daare?)

Problem Setter: E. AliMohammadi

A straight tunnel without branches is crowded with busy ants coming and going. Some ants walk left to right and others right to left. All ants walk at a constant speed of 1 cm/s. When two ants meet, they try to pass each other. However, some sections of the tunnel are narrow and two ants cannot pass each other. When two ants meet at a narrow section, they turn around and start walking in the opposite directions. When an ant reaches either end of the tunnel, it leaves the tunnel. The tunnel has an integer length in centimeters. Every narrow section of the tunnel is integer centimeters distant from the both ends. Except for these sections, the tunnel is wide enough for ants to pass each other. All ants start walking at distinct narrow sections. No ants will newly enter the tunnel. Consequently, all the ants in the tunnel will eventually leave it. Your task is to write a program that calculates the summation of all distances walked by the ants.

Figure shows the movements of the ants during the first two seconds in a tunnel 6 centimeters long. Initially, three ants, numbered 1, 2, and 3, start walking at narrow sections, 1, 2, and 5 centimeters distant from the left end, respectively. After 0.5 seconds, the ants 1 and 2 meet at a wide section, and they pass each other. Two seconds after the start, the ants 1 and 3 meet at a narrow section, and they turn around. Following figure corresponds to the first dataset of the sample input.



Input

The input consists one or more datasets. Integer t at the first line indicates the number of datasets.

There is a string s in each dataset which shows the initial position of ants. s_i indicates the initial state of tunnel at the position i . ' $<$ ' correspond to an ant which is going to left, ' $>$ ' correspond to an ant which is going to right and '.' means there is no ant at that position.

It's guaranteed that ($|s| \leq 1000$ and $t \leq 100$)

Output

For each dataset, print an integer: the summation of total distance walked by the ants, until they exit.

Sample Input	Sample Output
5 .><..<.> .<..... ><	12 0 1 0 2

Sample tests are written in larger font size for better visualization.



Problem M – Concert

Problem Setters: R. Shiri & M. Mohseni

Iran is the country of art and music. There are many Iranian poets and musicians in the world. But none of them are better than Reza Shiri. He is our most favorite singer in SBU.

Not only good at singing, Reza is also good at algorithms and mathematics. He has created a problem for Newbies2018 and if you solve it. You may win a ticket to his concert (I wish I had this chance).

A number is prime-lover, if sum of its digits in base-3 is a prime number. For example 2, 6, and 31 are prime-lovers, sum of digits of all these numbers are prime.

$$2 = (2)_3, \quad 6 = (20)_3, \quad 31 = (1011)_3$$

Checking whether a number is prime-lover or not is too easy task for you to win a ticket. So, I changed his problem a little.

Given two integers N, K. Can you calculate K'th smallest prime-lover which is not greater than N?

Input

The first line of input indicates the number of test cases (There will be at most 1000 test cases)

Each test case consists of two space-separated integers N, K. ($1 \leq N, K \leq 10^{13}$)

Output

For each test case, print the answer to the problem. If there is no such number, print -1.

Sample Input	Sample Output
3 10 3 10 6 10 7	5 10 -1

Explanation

Prime-Lover Numbers not greater than 10 are:

2, 4, 5, 6, 7, 10

$$2 = (2)_3$$

$$4 = (11)_3$$

$$5 = (12)_3$$

$$6 = (20)_3$$

$$7 = (21)_3$$

$$10 = (101)_3$$