Problem A. Christmas Tree

Program:	`tree.(cpp|java)`
Input:	`tree.in`
Balloon Color:	`Pink`

Christmas time is almost here! Families are getting ready for this by buying decorated evergreen trees. The christmas tree contains n nodes numbered from 0 to $n-1$ and rooted at node 0. Alice and Bob are playing around with the new tree and they will play a game using the tree in order to kill their boredom. Alice will carry a coloring marker and Bob will call out two types of instructions:

- $+x$, this means that Alice will color the node numbered x.
- $-x$, this means that Alice will clear the color on the node numbered x.

After each of these instructions, Alice should call out the lowest common ancestor (see note for definition) of all the colored nodes so far. Can you help Alice to reply to Bob's queries as fast as possible ?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 100$). Followed by **T** test cases. The first line of each test cases will contain one integer **N**, the number of nodes in the tree ($1 \leq N \leq 10^5$). The following **N - 1** lines will each contain a pair of integers **x** and **y** separated by a single space ($0 \leq x, y \leq N-1$) which means that node x is connected to node y. It's guaranteed that the given edges will form a tree. The following line will contain one integer **Q**, the number of instructions Bob will say ($1 \leq Q \leq 4 \times 10^5$). The following **Q** lines will each contain an instruction that Bob called out in the format of q_i a_i where $q_i \in \{+ , -\}$ and ($0 \leq a_i \leq N-1$). It's guaranteed that the clear instruction will only be applied on a colored node and the color instruction on an uncolored node)

Output

For each test instruction that Bob calls out, print a single line containing an integer corresponding to Alice's response to what is the lowest common ancestor of all the colored nodes or -1 if there are no colored nodes.

Examples

tree.in	standard output
1 10 1 4 5 4 1 0 6 8 6 1 1 3 7 6 9 7 9 2 7 + 2 + 8 + 0 - 0 + 9 - 9 + 1	2 6 0 6 6 6 1

Note

In graph theory and computer science, the lowest common ancestor (LCA) of two nodes v and w in a tree or directed acyclic graph (DAG) is the lowest (i.e. furthest from the root) node that has both v and w as descendants, where we define each node to be a descendant of itself.

Problem B. The Lion King

Program:	`lion.(cpp|java)`
Input:	`lion.in`
Balloon Color:	`Blue`

In the Pride Lands of Africa, a lion rules over the animals as king. The birth of King Mufasa and Queen Sarabi's son Simba creates envy and resentment in Mufasa's younger brother, Scar, who knows his nephew now replaces him as heir to the throne.

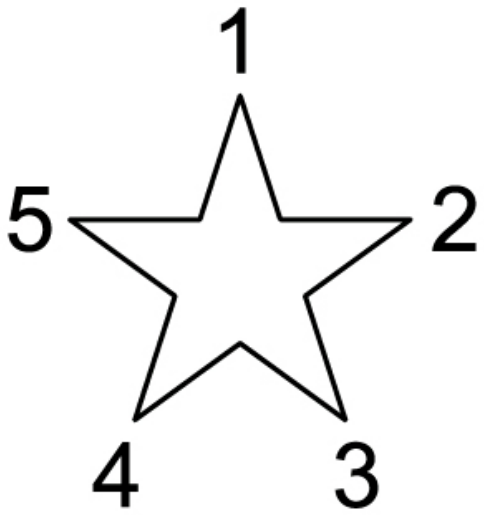
After Simba has grown into a young cub, Mufasa gives him a tour of the Pride Lands, teaching him the responsibilities of being a king and the circle of life. They spent the whole day in this tour and now it's time to sleep.

As Simba is still young, he didn't manage to sleep quickly and kept nudging his father to speak with him. Mufasa is really tired and would like to sleep so he thought of asking Simba a hard question to keep him busy. Mufasa asked Simba how many stars are there in the sky?

Simba sees the sky as an infinite 2D grid with some glowing points.

A star is a set of five points (p_1, p_2, p_3, p_4, and p_5) satisfying these conditions:

- $p_1.y > p_5.y$
- $p_5.y = p_2.y$
- $p_3.y, p_4.y < p_5.y$
- $p_5.x < p_1.x < p_2.x$
- $p_5.x < p_4.x < p_1.x$
- $p_1.x < p_3.x < p_2.x$



Can you help Simba answer this question as soon as possible?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 200$), followed by **T** test cases. The first line of each test case will contain one integer **N** ($5 \leq N \leq 5,000$) where N is the number of points in the sky.

The following **N** lines will each contain a pair of integers **x** and **y** separated by a single space ($-5,000 \leq x, y \leq 5,000$) representing the coordinates of the points.

Output

For each test case, print one line which contains the number of stars that Simba can see in the sky **modulo 1,000,000,007**. Each point can belong to multiple stars.

Examples

lion.in	standard output
2 5 0 5 4 4 -4 4 2 0 -2 0 8 0 5 4 4 -6 4 2 0 -1 0 -5 4 0 12 3 0	1 8

Problem C. Cultivating Mars

Program:	`mars.(cpp|java)`
Input:	`mars.in`
Balloon Color:	`Red`

Mars mission APOLLO was hit by a strong storm and astronaut Mark Watney got lost after being injured. Trying to save the rest of the crew, commander Melissa left Mark behind. The commander did so under the strong impression that Mark can never be alive after being hit. Luckily, Mark survived the injury and is now on his own on Mars. Unlike what people might think, Mark is really enjoying his time on Mars. The scenery is amazing and no body is there to annoy him.

After suffering for some time, Mark succeeded in contacting NASA and he is counting on their help to survive. He needs to stay alive on his own for 4 years until the next Mars mission rescues him. Mark likes it on Mars, but he has got some problems. The major one is food. He has food supplies that can keep him alive for only 5 months. Since he is a great botanist, Mark decided to grow crops on Mars. With NASA's help, he figured out how to do so, but one problem remains. He discovered that the field where he will grow the crops has the shape of a convex polygon. He modeled the field as a convex polygon in a 2D coordinate system, where the field contains a number of cells. The cells of the field are all the integer points inside and on the boundaries of the polygon (including the vertices). Additionally, the vertices of the field are integer coordinates. He can grow crops in only one cell in the field; the problem is which cell to choose. Mark discovered that depending on different factors on Mars, some cells can grow crops faster than other cells. With his great scientific abilities, he figured out a formula for this. He was able to represent these factors by a pair of integers $f = (a, b)$ and called it the **growing factor**. Depending on the growing factor $f = (a, b)$, each cell (x, y) in the field will have a **growing ability** of $g = a \times x + b \times y$. Mark wants to choose the cell with the maximum growing ability. He might cultivate the field more than once, and each time he might encounter a different growing factor (according to the time of the year he is cultivating the field). Hence, for each different growing factor Mark will encounter (or expect to encounter), he wants to know the maximum growing ability he can get from the cells of the field. Mark sent all the information to NASA to help him. NASA hired you to help them find the maximum cell growing ability Mark can get for each given growing factor.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer $\mathbf{T}$, the number of test cases ($1 \leq T \leq 100$), followed by $\mathbf{T}$ test cases. The first line of each test case will contain two integers $\mathbf{N}$ ($3 \leq N \leq 10,000$) and $\mathbf{Q}$ ($1 \leq Q \leq 100,000$) separated by a single space, where N is the number of vertices of the convex polygon representing the field and Q is the number of different growing factors Mark expects to encounter for that field. The following $\mathbf{N}$ lines will each contain a pair of integers $\mathbf{x}$ and $\mathbf{y}$ separated by a single space ($-10^9 \leq x, y \leq 10^9$) representing the coordinates of the vertices. The vertices will be given in anti-clockwise order. It's guaranteed that no three vertices will be on the same line. The following $\mathbf{Q}$ lines will each contain a pair of integers $\mathbf{a}$ and $\mathbf{b}$ ($-10^9 \leq a, b \leq 10^9$), separated by a single space, representing a growing factor.

Output

For each growing factor, print one line which contains the maximum growing ability Mark can get for that growing factor on the cells of the given convex polygon (field).

Examples

mars.in	standard output
1 3 2 -1 -1 3 1 1 3 1 1 -1 -1	4 2

Problem D. Fractionstellar

Program:	`fraction.(cpp|java)`
Input:	`fraction.in`
Balloon Color:	`Green`

It's 2050. Humans have already colonized Mars and other planets long time ago and there are already some programs for travelling to the other galaxies using wormholes. Scientists are currently studying the mysteries of the black holes. Their observations concluded that everything we know about physics and mathematics is completely different inside the black hole. For example, do you remember the greatest common divisor (GCD) and the lowest common multiple (LCM)? These functions are normally defined only on integers. The situation is different inside the black hole; GCD and LCM are also defined on rational numbers. For two rational numbers a/b and c/d: their GCD is the greatest rational number that divides both numbers to an integer, and their LCM is the lowest rational number that both numbers divide to an integer. For example, GCD(1/2, 1/3) = 1/6 and LCM(1/2, 1/3) = 1/1. Can you help the scientists in their missions solving out the mysteries of the black holes? Given two rational numbers, find their GCD and LCM inside the black hole.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 1000$). Followed by **T** test cases. Each test case contains four integers **a**, **b**, **c**, and **d** ($1 \leq a, b, c, d \leq 2 \times 10^9$) representing the two rational numbers a/b and c/d.

Output

For each test case, print a single line containing two rational numbers m/n and x/y, the GCD and the LCM of the two given rational numbers. m/n and x/y must be in their simplest form. In other words, the GCD(m,n) and GCD(x,y) must be 1.

Examples

fraction.in	stdout
2 1 2 1 3 1 5 1 7	1/6 1/1 1/35 1/1

Problem E. The Minions Quiz

Program:	`quiz.(cpp|java)`
Input:	`quiz.in`
Balloon Color:	`Black`

The minions have finally found their new master. This time, he is a Math professor and he is trying very hard to teach them math. He has been teaching them bitwise operators for over a year! They learnt about **AND(&)** and **OR (|)** operators and it is time for a quiz to test them.

The quiz is very simple, they will be given a number **A** of **AND(&)** operators, a number **B** of **OR (|)** operators and **(A + B + 1)** integers. They have to find the maximum number that can be obtained by inserting the & and | operators between the given nonnegative integers without changing their order.

Finally, there is a special requirement for this quiz, they are required to evaluate the operators from left to right.

Input

The first line of the input will be a single integer **T**, the number of test cases **($1 \leq T \leq 100$)**, followed by **T** test cases.

Each test case will consist of 2 lines. The first line will contain 2 integers **A** and **B ($0 \leq A, B \leq 10,000$)** representing the number of **AND(&)** and **OR (|)** operators, respectively. The second line of input will consist of **(A + B + 1)** 64-bit nonnegative integers separated by single spaces.

Output

For each test case, output a single line containing the maximum number that can be obtained by inserting the operators between the given integers.

Examples

quiz.in	standard output
2 1 1 1 4 5 2 2 2 3 11 4 5	5 7

Problem F. Uberfication

Program:	`uber.(cpp|java)`
Input:	`uber.in`
Balloon Color:	`Purple`

Salem decided to launch a new service for carpooling to connect drivers with empty seats to people travelling the same way. Riders are charged small amounts of money compared to other means of transportation and drivers can earn money without wasting time. The ride fee is mainly dependent on the distance between the trip starting point and the destination. In order to beat competitors in the same market, Salem decided to make the service free of charge between any two points having more than one path to reach one another. For simplicity, we can assume that the city in which he will launch his service is modeled as an undirected graph having N nodes and M edges. All edges have the same length of 1 unit distance. A path is a sequence of edges connecting a sequence of nodes which are all distinct from one another. Salem wants your help to give primary estimates for the revenue of his service by calculating the distance between all pairs of nodes which have a unique way to reach one another and ignoring all pairs having more than one way between each other. In your revenue analysis, you must count each pair only once.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 50$). Followed by **T** test cases. The first line of each test case will contain two integers **N** ($1 \leq N \leq 50,000$) and **M** ($0 \leq M \leq 150,000$), the number of nodes and edges in the city. They are separated by a single space. The following **M** lines will each contain a pair of integers **x** and **y** ($1 \leq x, y \leq N$), separated by a single space, which means that node x is connected to node y. It's guaranteed that the input will not contain more than one edge between any two nodes or an edge from a node to itself.

Output

For each test case, print a single line containing an integer representing the total distances between all pairs of nodes having a unique way to reach one another.

Examples

uber.in	standard output
2 5 5 1 2 2 3 3 1 2 4 1 5 4 3 1 2 2 3 2 4	2 9

Note

WARNING: large input data, be careful with certain languages.

Problem G. DNA Evolution

Program:	`dna.(cpp|java)`
Input:	`dna.in`
Balloon Color:	`White`

Fox Ciel just had a new baby born (Fox Jaro). Fox Ciel has always been interested in genetics and DNA sequences and he would like to apply what he knows on his son! A DNA consists of a sequence of nucleotides. There are four types of nucleotides: A, C, G, and T. DNA sequences are passed from a parent to his children, however each newborn fox gets one more nucleotides than his father (the new nucleotide is attached to the beginning of the sequence). For example, Fox Ciel's DNA is CACAA and his son Jaro's DNA is ACACAA (notice the A, that Jaro acquired more than Ciel, attached to the DNA inherited from Ciel).

Ciel was thinking about his ancestors and how his new born looks like them. He noticed that the similarity of his son and one of his grand parents is proportional to the maximum number of common characters they have as a common prefix and he named it as the similarity coefficient. He wrote down all ancestors' DNA and calculated the similarity coefficient between each one of them and Jaro's DNA (ACACAA), then wrote that down on a note similar to the following table. (see note for a formal definition)

ACACAA	6
CACAA	0
ACAA	3
CAA	0
AA	1
A	1

Rabbit Hanako messed up with Ciel's note and deleted Jaro's DNA sequence. If Ciel finds this out, he'll attack Hanako and eat him! Can you help Hanako restore the DNA sequence as soon as possible before Ciel finds this out?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 100$). Followed by **T** test cases.

Each test case will be presented on a single line by integers separated by single spaces. The first integer on this line will denote the number of characters of the DNA sequence **N** followed by **N** integers a_i denoting the similarity coefficient at position i for all $0 \leq i < n$ (as shown in the table), where ($1 \leq N \leq 10^5$) and ($0 \leq a_i \leq 10^5$).

Output

For each test case, print a single line containing **S** which is the DNA sequence. If there are multiple valid sequences, print the least lexicographical one. If there are no valid sequences and Hanako can't restore the sequence, print "Impossible"(without the quotes).

String x is lexicographically less than string y, if either x is a prefix of y or there exists such i where ($1 \leq i \leq min(|x|, |y|)$), $x_i < y_i$ and for any j ($1 \leq j < i$) $x_j = y_j$.

Examples

dna.in	standard output
3 1 1 3 3 0 1 3 3 3 3	A ACA Impossible

Note

Formally, given a string S (DNA) of length n, the note will contain an array A where A[i] is the length of the longest substring starting from S[i] which is also a prefix of S. In other words, the maximum k such that S[j] = S[i + j] for all $0 \leq j < k$.

Problem H. Bingo!

Program:	`bingo.(cpp|java)`
Input:	`bingo.in`
Balloon Color:	`Yellow`

The new academic year has just started! Everyone is super excited about learning new stuff in Hogwarts. Harry Potter is attending a new class about spells, where he is learning how to read people's minds and determine their next actions. Harry is going to play a game with Hermione and he would like to try using the new spell to win the game.

Harry and Hermione are going to play Bingo! Each player will have a 5×5 board that contains the numbers from 1 to 25 (each number appearing exactly once). The game goes in turns and Harry starts. Each player says a number (that was not marked yet) out loud and both players mark this number on their boards. The first player to fill up 5 complete rows, columns, or diagonals shouts Bingo! and wins the game. If both of them shout at the same time, it will be considered a tie and no one wins.

As Harry can read Hermione's mind, he can see her board configuration and can know what is the sequence of numbers she is going to shout. This shall be a permutation of the 25 numbers that she is going to call out (of course she will skip numbers that Harry already shouted!). Ron can not leave them playing alone! He asked Harry a mathematics question. In how many ways can you win this game? Two ways are considered different if at the i-th turn he would shout a different number. As this number might be large, print it modulo 1,000,000,007.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 50$), followed by **T** test cases.

Each test case will be presented on multiple lines. The first 5 lines represent Harry's board, where each line contains 5 integers separated by single spaces. The second 5 lines represent Hermione's board in the same format. The last line will contain a permutation of the numbers from 1 to 25 separated by single spaces representing Hermione's shout outs. It's guaranteed that each number from 1 to 25 will appear only once in Harry's and Hermione's boards and in Hermione's shout outs.

Output

For each test case, print a single line containing the number of ways Harry can win the game modulo 1,000,000,007.

Examples

bingo.in	standard output
2 16 22 11 21 15 18 24 13 14 25 8 9 7 2 4 3 19 23 17 12 20 5 1 6 10 24 17 21 8 10 15 2 16 25 7 13 12 3 14 9 19 1 6 4 20 23 22 5 11 18 15 18 16 17 13 19 1 24 23 21 20 4 9 11 2 12 22 7 8 3 25 14 5 6 10 20 11 22 2 19 17 25 23 14 13 10 8 6 21 18 12 1 5 24 16 7 4 15 3 9 18 10 20 12 11 1 23 13 8 15 6 7 9 19 24 25 21 3 17 2 14 22 4 16 5 9 13 6 24 5 21 10 22 25 11 23 15 17 8 19 2 20 18 7 4 16 12 1 3 14	775283109 200384431

Problem I. Journey

Program:	`journey.(cpp|java)`
Input:	`journey.in`
Balloon Color:	`Cyan`

One day, Homer was bored in his house and decided to go in a journey to discover the lands of Springfield. The lands of Springfield is an infinite grid. Homer's house is located at cell $(0, 0)$ and his journey consisted of N steps, where each step is either move one cell right or one cell down.

Being bored already, Homer didn't want his journey to be boring as well. He decided he won't move in the same direction for more than K consecutive steps. Thus, a journey is considered to be interesting if for each K+1 consecutive steps Homer has moved in both directions.

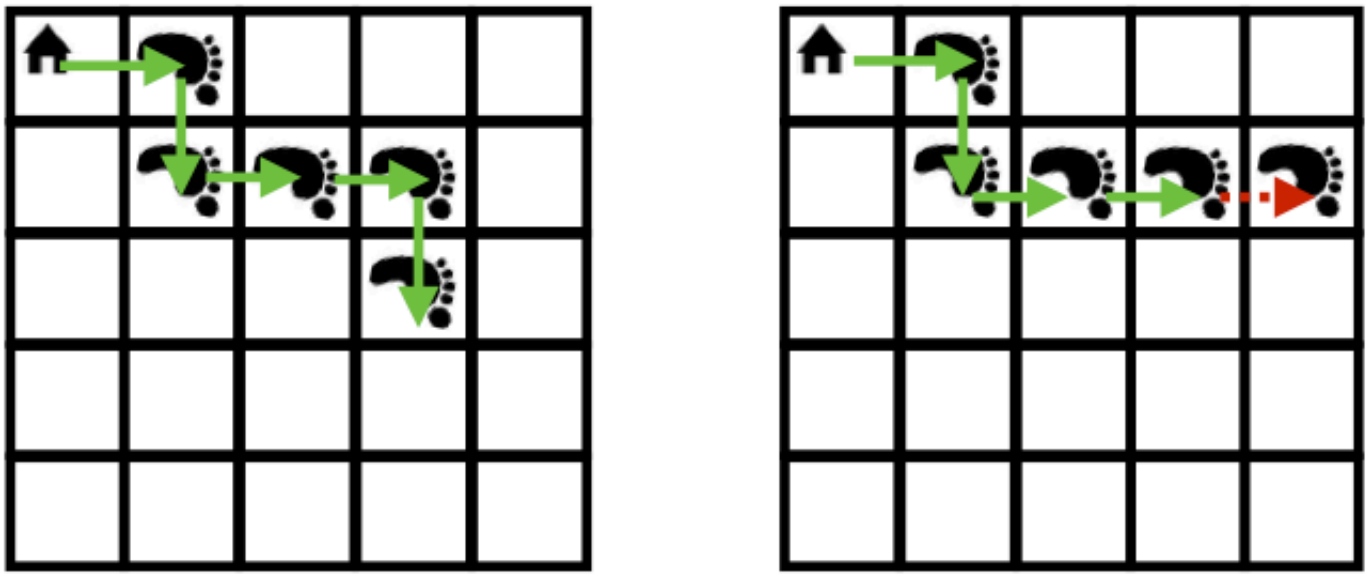


Figure 1: Example with N=5 and K=2 (first test case).

Given N and K, count the number of interesting journeys Homer can make. Two Journeys are considered different if for some i the i^{th} step in the first Journey differs from that of the second Journey. Since the number can be large, print it modulo 1,000,000,007.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 500$), followed by **T** test cases.

Each test case will be presented on a single line containing two integers separated by a single space. The first integer will denote the number of steps in Homer's journey **N**, followed by the second integer **K** representing the maximum number of consecutive steps Homer can take while moving in the same direction, where ($0 \leq N \leq 10^5$) and ($0 \leq K \leq 10^5$).

Output

For each test case, output a single line denoting the number of different journeys Homer can make modulo 1,000,000,007.

Examples

journey.in	standard output
2 5 2 10 1	16 2

Problem J. Banana

Program:	`banana.(cpp|java)`
Input:	`banana.in`
Balloon Color:	`Dark Green`

After NASA discovered water on Mars, they decided to expand their exploration hoping to find some alien intelligence on the planet. After months of exploration, they were actually surprised to find out that the planet has been inhabited by minions.

NASA started communications with the minions and the first message they received was **Mo amo Banana**. At first, it was really hard to decipher the message but after sometime they managed to workout a dictionary that maps English words to Minionese words. You are going to help NASA build the translator to ease their communication with Minions for the good and prosperity of mankind and minionkind.

Input

The first line of input will contain a single integer **N**, the number of words in the dictionary ($1 \leq N \leq 100$). The following **N** lines will each contain a sentence of the format $x = y$ where x is an English word and y is a Minionese word. The next line will contain an integer **T**, the number of test cases ($1 \leq T \leq 100$). Each test case will start with a line containing an integer **K**, the number of words in the sentence ($1 \leq K \leq 100$) and the next line will contain **K** space separated English words. All the English words in the test cases exist in the defined dictionary. Also, all the words consist only of English alphabet, and will be at most 20 characters long.

Output

For each test case, print a single line containing the space separated Minionese words after translation.

Examples

banana.in	standard output
4 I = mo love = amo icecream = gelatooo banana = banana 2 3 I love banana 3 I love icecream	mo amo banana mo amo gelatooo

Problem K. Road Network

Program:	`batman.(cpp|java)`
Input:	`batman.in`
Balloon Color:	`Dark Blue`

After a fierce battle with his opponent, Bruce Wayne finally won the elections and became the mayor of Gotham. Like every other politician, he had an agenda with lots of projects for the sake of Gotham's prosperity, but he was met with the same problem, lack of fund.

He decided to tackle the problem from a different perspective; he will allow companies to buy roads in the city (roads in the city are undirected). The city will get the money needed for the projects and the companies can use the roads for advertisements (or so he thought).

After the deal was done, the companies were more cunning than he expected. They started to threaten that they will block exactly one road in the city and prevent people from getting to their work, in the hope that people will revolt against Mayor Wayne. The problem was that the city is designed as a tree of connected zones, where there is only one unique path between any two zones. Hence, blocking a road means that some zones are not reachable from others anymore.

Mayor Wayne discussed the problem with his council and identified what they called vulnerable roads. A road is vulnerable if blocking it can disconnect two zones from each other. Mayor Wayne wants to prevent this from happening by building more roads but his budget could afford building only one extra road. Can you help him figure out which road he should build, such that he minimizes the number of vulnerable roads?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 100$) followed by **T** test cases. The first line of each test case will contain one integer **N**, the number of zones in the city ($1 \leq N \leq 10,000$). The following **N - 1** lines will each contain a pair of integers **x** and **y** separated by a single space ($1 \leq x, y \leq N$) which means that zone x is connected to zone y. It's guaranteed that the edges will form a tree.

Output

For each test case, print a single line containing an integer, the minimum number of vulnerable roads in the city after building the new road.

Examples

batman.in	standard output
2 3 1 2 1 3 4 1 2 2 3 2 4	0 1