

The 2019 ICPC China Shaanxi Provincial Programming Contest



Contest Session

June 2, 2019



Problem List

A	Digit Mode
B	Grid with Arrows
C	0689
D	Pick Up
E	Turn It Off
F	K-hour Clock
G	Paper-cutting
H	To the Park
I	Unrooted Trie
J	Coolbits
K	Escape Plan
L	Digit Product

This problem set should contain 12 (twelve) problems on 14 (fourteen) numbered pages. Please inform a runner immediately if something is missing from your problem set.

Prepared by Zhejiang University Programming Contest Problem Setter Team.

<http://acm.zju.edu.cn/pcpst/>

Hosted by



西安电子科技大学
XIDIAN UNIVERSITY

Sponsored by



Problem Set Prepared by



Problem A. Digit Mode

Let $m(x)$ be the *mode* of the digits in decimal representation of positive integer x . The mode is the largest value that occurs most frequently in the sequence. For example, $m(15532) = 5$, $m(25252) = 2$, $m(103000) = 0$, $m(364364) = 6$, $m(114514) = 1$, $m(889464) = 8$.

Given a positive integer n , DreamGrid would like to know the value of $(\sum_{x=1}^n m(x)) \bmod (10^9 + 7)$.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains a positive integer n ($1 \leq n < 10^{50}$) without leading zeros.

It's guaranteed that the sum of $|n|$ of all test cases will not exceed 50, where $|n|$ indicates the number of digits of n in decimal representation.

Output

For each test case output one line containing one integer, indicating the value of $(\sum_{x=1}^n m(x)) \bmod (10^9 + 7)$.

Example

standard input	standard output
5	45
9	615
99	6570
999	597600
99999	5689830
999999	

Problem B. Grid with Arrows

BaoBao has just found a grid with n rows and m columns in his left pocket, where the cell in the j -th column of the i -th row (indicated by (i, j)) contains an arrow (pointing either upwards, downwards, leftwards or rightwards) and an integer $a_{i,j}$.

BaoBao decides to play a game with the grid. He will first select a cell as the initial cell and tick it. After ticking a cell (let's say BaoBao has just ticked cell (i, j)), BaoBao will go on ticking another cell according to the arrow and the integer in cell (i, j) .

- If the arrow in cell (i, j) points upwards, BaoBao will go on ticking cell $(i - a_{i,j}, j)$ if it exists.
- If the arrow in cell (i, j) points downwards, BaoBao will go on ticking cell $(i + a_{i,j}, j)$ if it exists.
- If the arrow in cell (i, j) points leftwards, BaoBao will go on ticking cell $(i, j - a_{i,j})$ if it exists.
- If the arrow in cell (i, j) points rightwards, BaoBao will go on ticking cell $(i, j + a_{i,j})$ if it exists.

If the cell BaoBao decides to tick does not exist, or if the cell is already ticked, the game ends.

BaoBao is wondering if he can select a proper initial cell, so that he can tick every cell in the grid exactly once before the game ends. Please help him find the answer.

Input

There are multiple test cases. The first line contains an integer T , indicating the number of test cases. For each test case:

The first line contains two integers n and m ($1 \leq n \times m \leq 10^5$), indicating the number of rows and columns of the grid.

For the following n lines, the i -th line contains a string s_i consisting of lowercased English letters ($|s_i| = m$, $s_{i,j} \in \{\text{'u' (ascii: 117), 'd' (ascii: 100), 'l' (ascii: 108), 'r' (ascii: 114)}\}$), where $s_{i,j}$ indicates the direction of arrow in cell (i, j) .

- If $s_{i,j} = \text{'u'}$, the arrow in cell (i, j) points upwards.
- If $s_{i,j} = \text{'d'}$, the arrow in cell (i, j) points downwards.
- If $s_{i,j} = \text{'l'}$, the arrow in cell (i, j) points leftwards.
- If $s_{i,j} = \text{'r'}$, the arrow in cell (i, j) points rightwards.

For the following n lines, the i -th line contains m integers $a_{i,1}, a_{i,2}, \dots, a_{i,m}$ ($1 \leq a_{i,j} \leq \max(n, m)$), where $a_{i,j}$ indicates the integer in cell (i, j) .

It's guaranteed that the sum of $n \times m$ of all test cases does not exceed 10^6 .

Output

For each test case output one line. If BaoBao can find a proper initial cell, print "Yes" (without quotes), otherwise print "No" (without quotes).

Example

standard input	standard output
2	Yes
2 3	No
rdd	
url	
2 1 1	
1 1 2	
2 2	
rr	
rr	
1 1	
1 1	

Note

For the first sample test case, BaoBao can select cell (1, 2) as the initial cell, so that he can tick all the cells exactly once in the following order: (1, 2), (2, 2), (2, 3), (2, 1), (1, 1), (1, 3).

For the second sample test case, BaoBao can only tick at most 2 cells no matter which cell is selected as the initial cell.

Problem C. 0689

We call a string as a 0689-string if this string only consists of digits '0', '6', '8' and '9'. Given a 0689-string s of length n , one **must** do the following operation exactly once: select a non-empty substring of s and rotate it 180 degrees.

More formally, let s_i be the i -th character in string s . After rotating the substring starting from s_l and ending at s_r 180 degrees ($1 \leq l \leq r \leq n$), string s will become string t of length n extracted from the following equation, where t_i indicates the i -th character in string t :

$$t_i = \begin{cases} s_i & \text{if } 1 \leq i < l \text{ or } r < i \leq n \\ '0' & \text{if } l \leq i \leq r \text{ and } s_{l+r-i} = '0' \\ '6' & \text{if } l \leq i \leq r \text{ and } s_{l+r-i} = '9' \\ '8' & \text{if } l \leq i \leq r \text{ and } s_{l+r-i} = '8' \\ '9' & \text{if } l \leq i \leq r \text{ and } s_{l+r-i} = '6' \end{cases}$$

What's the number of different strings one can get after the operation?

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first and only line contains a 0689-string s ($1 \leq |s| \leq 10^6$).

It's guaranteed that the sum of $|s|$ of all test cases will not exceed 10^7 .

Output

For each test case output one line containing one integer, indicating the number of different strings one can get after applying the operation exactly once.

Example

standard input	standard output
2	8
0689	2
08	

Note

We hereby explain the first sample test case.

Substring	Result	Substring	Result
0	0689	68	0899
6	0989	89	0668
8	0689	068	8909
9	0686	689	0689
06	9089	0689	6890

It's easy to discover that we can get 8 different strings after the operation.

Problem D. Pick Up

Grid City is a city on an infinite two-dimensional plane, where for all $k \in \mathbb{Z}$ (\mathbb{Z} is the set of all integers) lines $x = k$ and $y = k$ are the streets of the city. People can only move along the roads to go from one position to another. That's why the city is called the Grid City!

Two friends, BaoBao and DreamGrid, live happily in the city. Today BaoBao is heading from his home positioned at (x_A, y_A) ($x_A, y_A \in \mathbb{Z}$) towards the shopping mall positioned at (x_C, y_C) ($x_C, y_C \in \mathbb{Z}$). However, it's too far for him to walk there, so he decides to call DreamGrid whose home is positioned at (x_B, y_B) ($x_B, y_B \in \mathbb{Z}$) for help.

BaoBao and DreamGrid set out separately from their homes at the same time. Different from BaoBao who walks at a speed of a units per minute, DreamGrid drives a car and moves at a speed of b units per minute. When DreamGrid and BaoBao meet at the same point, DreamGrid can pick up BaoBao and they can then move at a speed of b units per minute together. It takes no time to turn around or pick up BaoBao.

What's the minimum time needed for BaoBao to go from his home to the shopping mall? Note that it's NOT necessary for DreamGrid to pick up BaoBao if it will be faster for BaoBao to get to the destination.

Input

There are multiple test cases. The first line of the input contains an integer T ($1 \leq T \leq 10^5$), indicating the number of test cases. For each test case:

The first line contains two integers a and b ($1 \leq a < b \leq 10^9$), indicating the walking speed of BaoBao and the driving speed of DreamGrid.

The second line contains six integers x_A, y_A, x_B, y_B, x_C and y_C ($-10^9 \leq x_A, y_A, x_B, y_B, x_C, y_C \leq 10^9$), indicating the position of BaoBao's home, DreamGrid's home and the shopping mall. It's guaranteed that these three points are different from each other.

Output

For each test case output one line containing one number, indicating the shortest time for BaoBao to arrive at the shopping mall. Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6} .

Example

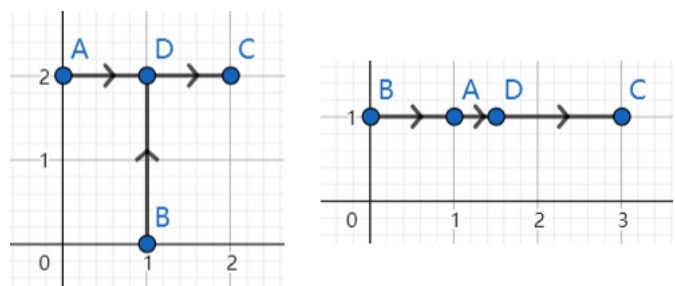
standard input	standard output
3	1.5000000000000000
1 2	1.0000000000000000
0 2 1 0 2 2	2.0000000000000000
1 3	
1 1 0 1 3 1	
1 2	
0 0 100 100 1 1	

Note

The sample test cases are shown on the right.

For the first sample test case, BaoBao and DreamGrid will meet at $D(1,2)$ and then DreamGrid drives BaoBao to the shopping mall.

For the second sample test case, BaoBao and DreamGrid will meet at $D(1.5,1)$ and then DreamGrid drives BaoBao to the shopping mall.



Problem E. Turn It Off

It's already 21:30 now, and it's time for BaoBao to go to bed. To ensure his sleeping quality, BaoBao decides to turn all the lights in his bedroom off.

There are n lights, numbered from 1 to n , arranged in a row in BaoBao's bedroom. Each time BaoBao can select an integer i and turn all the lights numbered from i to $(i + L - 1)$ (both inclusive) off, where L is a predefined positive integer. Note that each time the value of L must be the same.

Given the initial status of all the lights, please help BaoBao determine the smallest possible L so that he can turn all the lights off within k times.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains two integers n and k ($1 \leq k \leq n \leq 2 \times 10^5$).

The second line contains a string s ($|s| = n$, $s \in \{ '0', '1' \}$) indicating the initial status of the lights. Let s_i be the i -th character in s , if $s_i = '1'$ then the i -th light is initially on, otherwise it's initially off. It's guaranteed that there is at least one '1' in s .

It's guaranteed that the sum of n of all test cases will not exceed 2×10^6 .

Output

For each test case output one line containing one integer, indicating the smallest possible L .

Example

standard input	standard output
2	3
10 4	1
0101011111	
3 1	
010	

Problem F. K-hour Clock

A “ k -hour clock” is a day keeping method which follows the rules below:

- A day is divided into k hours, where the i -th hour is called the $(i - 1)$ o’clock;
- If it’s x o’clock now, it will be $(x + 1)$ o’clock after 1 hour if $0 \leq x < k - 1$;
- If it’s $(k - 1)$ o’clock now, it will be 0 o’clock after 1 hour.

We know that it’s x o’clock now, and after y hours it will be z o’clock. What’s the value of k ?

Input

There are multiple test cases. The first line of the input is an integer T (about 10^5), indicating the number of test cases. For each test case:

The first and only line contains three integers x , y and z ($0 \leq x, z \leq 10^9$, $1 \leq y \leq 10^9$).

Output

For each test case output one line containing one integer, indicating the value of k . Note that there must be $1 \leq k \leq 2 \times 10^9$. If there are multiple valid answers, you can print any of them; If there is no valid answer, print “-1” (without quotes) instead.

Example

standard input	standard output
4	12
11 18 5	24
3 49 4	3
1 9 1	-1
1 3 10	

Problem G. Paper-cutting

Paper-cutting is one of the oldest and most popular folk arts in China. It can be geographically divided into a southern and a northern style. The southern style, represented by works from Yangzhou in Jiangsu Province and Yueqing in Zhejiang Province, features ingenious and beautiful designs, exquisite carving and interesting shapes. The northern style, mainly from Yuxian and Fengning in Hebei Province and best represented by works from northern Shaanxi, features exaggerated shapes, vigorousness, vivid depictions, and diverse patterns.

There are basic cut-outs, consisting of a single image, and symmetrical designs, that are usually created by some folding over a proportioned crease, and then cutting a shape, so that when unfolded, it forms a symmetrical design. Chinese paper cuttings are usually symmetrical. The paper cutouts are usually in an even number series of 2, 4, 24, etc.

You are given a piece of paper in the shape of a matrix of size $n \times m$ and it is partitioned into $n \times m$ blocks of size 1×1 . The piece of paper can be folded in the following way:

1. You choose a vertical line between two of its columns or a horizontal line between two of its rows. This line splits the paper into two sides.
2. You use the line as the folding axis and fold the smaller side of the paper onto the larger one going over the axis. If both sides of the paper are of equal size, you may fold from either side.

You would like to make a paper-cutting masterpiece from this paper. At first, you fold the paper using the method above several times (including zero times). Then you use scissors to cut the paper and each time you can cut out a connected component from the paper and throw it away. Note that two 1×1 blocks are connected if they share an edge.

Given the final look of the paper – a matrix of size $n \times m$ containing 0s and 1s, you would like to know the minimum number of cuts needed when using the scissors.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains two integers n and m ($1 \leq n \times m \leq 10^6$) – the size of the paper.

Each of the next n lines contains a binary string of length m , where '0' means the 1×1 block is cut out and '1' means the 1×1 block remains on the final paper-cutting.

It is guaranteed that the sum of $n \times m$ over all test cases does not exceed 10^6 .

Output

For each test case output one line containing one integer, indicating the minimum number of cuts needed.

Example

standard input	standard output
3	1
2 5	4
11001	0
11001	
5 7	
1001100	
0110011	
0101101	
0010010	
1000000	
3 2	
11	
11	
11	

Note

For the first sample test case, you can fold in the following way and cut the only 0 out:

$$\begin{array}{ccc|cc} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{array} \rightarrow \frac{\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 1 & 0 \end{array}}{\phantom{\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 1 & 0 \end{array}}} \rightarrow 1 \quad 1 \quad 0$$

For the second sample test case, you can fold in the following way and cut the 4 connected components of 0s out:

$$\begin{array}{cccc|cccc} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & \rightarrow 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}$$

Problem H. To the Park

BaoBao and his $(n - 1)$ classmates are going to the park. For convenience, their teacher DreamGrid has numbered the students from 1 to n and decides to form the students into some groups, where each group consists of exactly two students.

For some reason, DreamGrid requires that the indices of the two students in the same group should have a common divisor greater than 1. Note that each student can only belong to at most one group, and it's not necessary that every student belongs to a group.

Please help DreamGrid form as many groups as possible.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first and only line contains an integer n ($1 \leq n \leq 10^5$), indicating the number of students.

It's guaranteed that the sum of n of all test cases will not exceed 10^6 .

Output

For each test case output one line. The line first contains an integer k indicating the number of groups, then $2k$ integers a_1, a_2, \dots, a_{2k} follow, indicating that student a_1 and a_2 belong to the same group, student a_3 and a_4 belong to the same group, ..., student a_{2k-1} and a_{2k} belong to the same group. The integers in a line are separated by a space. If there are multiple valid answers, you can print any of them.

Please, DO NOT output extra spaces at the end of each line, or your solution may be considered incorrect!

Example

standard input	standard output
3	0
1	1 2 4
4	2 2 4 3 6
6	

Problem I. Unrooted Trie

Recall the definition of a trie:

- A trie of size n is a rooted tree with n vertices and $(n - 1)$ edges, where each edge is marked with a character;
- Each vertex in a trie represents a string. Let $s(x)$ be the string vertex x represents;
- The root of the trie represents an empty string. Let vertex u be the parent of vertex v , and let c be the character marked on the edge connecting vertex u and v , we have $s(v) = s(u) + c$. Here $+$ indicates string concatenation, not the normal addition operation.

We say a trie is valid, if the string each vertex represents is distinct.

Given an unrooted tree with n vertices and $(n - 1)$ edges, where each edge is marked with a character, how many different vertices can be selected as the root of the tree so that the tree becomes a valid trie?

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains an integer n ($1 \leq n \leq 10^5$), indicating the size of the tree.

For the following $(n - 1)$ lines, the i -th line contains two integers u_i, v_i ($1 \leq u_i, v_i \leq n$) and a character c_i separated by a space, indicating that there is an edge marked with a character c_i connecting vertex u_i and v_i . It's guaranteed that c_i will only be lower-case English letters.

It's guaranteed that the given graph is a tree, and the sum of n of all test cases will not exceed 10^6 .

Output

For each test case output one line containing one integer, indicating the number of different vertices that can be selected as the root of the tree to make it a valid trie.

Example

standard input	standard output
2	2
6	0
3 1 a	
3 2 a	
3 4 b	
4 5 c	
4 6 d	
6	
3 1 a	
3 2 a	
3 4 b	
5 4 c	
6 4 c	

Note

For the first sample test case, we can only select vertex 1 or vertex 2 as the root, otherwise $s(1)$ and $s(2)$ will be the same.

For the second sample test case, no matter which vertex we select as the root, $s(1)$ and $s(2)$, or $s(5)$ and $s(6)$ will be the same.

Problem J. Coolbits

Given n intervals $[l_1, r_1], [l_2, r_2], \dots, [l_n, r_n]$, one must select an integer from each of the intervals and calculate their bitwise and value b . What's the maximum possible b one can get?

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains an integer n ($1 \leq n \leq 10^5$), indicating the number of intervals.

For the following n lines, the i -th line contains two integers l_i and r_i ($0 \leq l_i \leq r_i \leq 10^9$), indicating the i -th interval.

It's guaranteed that the sum of n of all test cases will not exceed 10^6 .

Output

For each test case output one line containing one integer, indicating the maximum possible b one can get.

Example

standard input	standard output
2	6
3	100
0 8	
2 6	
3 9	
1	
1 100	

Note

For the first sample test case, one can select 7, 6 and 7 from the three intervals and get their bitwise and value 6.

Problem K. Escape Plan

BaoBao, one of the most famous monster hunters, wakes up in the middle of Heltion City dominated by monsters. Having troubles remembering what has happened, BaoBao decides to escape from this horrible city as soon as possible. Despite arming no weapon, he luckily puts his hand on a map in his right pocket, which contains valuable information that can possibly help him find a way out.

According to the map, Heltion City is composed of n spots connected by m undirected paths. Starting from spot 1, BaoBao must head towards any of the k exits of the city to escape, where the i -th of them is located at spot e_i .

However, it's not an easy task for BaoBao to escape since monsters are everywhere in the city! For all $1 \leq i \leq n$, d_i monsters are wandering near the i -th spot, so right after BaoBao arrives at that spot, at most d_i paths connecting the spot will be blocked by monsters and are unable for BaoBao to pass. When BaoBao leaves the i -th spot, the monsters will go back to their nests and the blocked paths are clear. Of course, if BaoBao comes back to the spot, at most d_i paths will be again blocked by the monsters. The paths blocked each time may differ.

As BaoBao doesn't know which paths will be blocked, please help him calculate the shortest time he can escape from the city in the worst case.

Input

There are multiple test cases. The first line of the input contains an integer T (about 100), indicating the number of test cases. For each test case:

The first line contains three integers n , m and k ($1 \leq n \leq 10^5$, $1 \leq m \leq 10^6$, $1 \leq k \leq n$), indicating the number of spots, the number of undirected paths and the number of exits of the city.

The second line contains k distinct integers e_1, e_2, \dots, e_k ($1 \leq e_i \leq n$), indicating k exits of Heltion City.

The third line contains n integers d_1, d_2, \dots, d_n ($0 \leq d_i \leq m$), where d_i indicates the number of monsters at the i -th spot.

For the following m lines, the i -th line contains three integers x_i, y_i and w_i ($1 \leq x_i, y_i \leq n$, $x_i \neq y_i$, $1 \leq w_i \leq 10^4$), indicating an undirected edge of length w_i connecting spot x_i and y_i .

It's guaranteed that the total sum of n will not exceed 10^6 and the total sum of m will not exceed 3×10^6 .

Output

For each case output one line containing one integer. If BaoBao can get to some exit in the worst case, output the shortest possible time cost; Otherwise if BaoBao cannot get to any exit in the worst case, output "-1" (without quotes) instead.

Example

standard input	standard output
2	4
3 4 1	-1
3	
1 1 1	
1 2 1	
1 2 2	
2 3 1	
2 3 2	
3 2 2	
2 3	
2 0 0	
1 2 1	
1 3 1	

Problem L. Digit Product

Define the “digit product” $f(x)$ of a positive integer x as the product of all its digits. For example, $f(1234) = 1 \times 2 \times 3 \times 4 = 24$, and $f(100) = 1 \times 0 \times 0 = 0$.

Given two integers l and r , please calculate the following value:

$$\left(\prod_{i=l}^r f(i)\right) \bmod (10^9 + 7)$$

In case that you don't know what \prod represents, the above expression is the same as

$$(f(l) \times f(l+1) \times \cdots \times f(r)) \bmod (10^9 + 7)$$

Input

There are multiple test cases. The first line of the input contains an integer T (about 10^5), indicating the number of test cases. For each test case:

The first and only line contains two integers l and r ($1 \leq l \leq r \leq 10^9$), indicating the given two integers. The integers are given without leading zeros.

Output

For each test case output one line containing one integer indicating the answer.

Example

standard input	standard output
2	362880
1 9	367416
97 99	

Note

For the first sample test case, the answer is $9! \bmod (10^9 + 7) = 362880$.

For the second sample test case, the answer is $(f(97) \times f(98) \times f(99)) \bmod (10^9 + 7) = (9 \times 7 \times 9 \times 8 \times 9 \times 9) \bmod (10^9 + 7) = 367416$.