2021“MINIEYE 杯”中国大学生算法设计超级联赛（7）

题目列表

Problem List

1001	Fall with Fake Problem
1002	Fall with Soldiers
1003	Fall with Trees
1004	Link with Balls
1005	Link with EQ
1006	Link with Grenade
1007	Link with Limit
1008	Smzzl with Greedy Snake
1009	Smzzl with Safe Zone
1010	Smzzl with Tropical Taste
1011	Yiwen with Formula
1012	Yiwen with Sqc

2021 年 8 月 10 日

Problem 1001. Fall with Fake Problem

Input file: `standard input`
Output file: `standard output`

A boy, whose ID is smzzl, loves solving problems on Codeforces. Some day, he was solving the following problem:

Given a string S, find the lexicographical-smallest string T, whose lexicographical order is **no smaller than** S, the times each letter appears in T are multiples of k, and $|S| = |T|$.

“The problem is too easy! I want a harder one!”

So, he modified the problem. He no longer requires the times each letter appears in T to be a multiple of k. Instead, he wants them to be a factor of k or 0.

The problem actually becomes harder, but not so much harder. Fall, the teammate of smzzl, solved it immediately! And now, Fall wants you to solve it, so he can check his answer.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 32)$ – the number of test cases.

For each test case, there are two lines.

The first line contains two integers $n, k (1 \le n, k \le 10^5)$, which are the length of S and the argument mentioned above.

The second line contains a string S, which only contains lower case letters.

It is guaranteed that the sum of $|S|$ over all test cases does not exceed 10^6.

Output

For each test case, output a string T in a line, which is your answer. If there is no such T, output ‘-1’ instead (without quotes).

Example

standard input	standard output
2 5 6 aaaaa 10 18 abcabcabcd	aaabb abcabcabcd

Problem 1002. Fall with Soldiers

Input file: `standard input`
Output file: `standard output`

Fall is playing a game about a war.

There are n soldiers standing in a line. Some of them belong to the player, while others belong to the enemy. Due to the spies, the belonging of some of them is unknown. Now, as an artillery, the player may win this game under the following rules after identifying the belonging of each soldier:

Step 1: Choose a soldier which belongs to the player. There should be exactly two soldiers next to the chosen one, which means the chosen soldier should be neither at the beginning nor at the end of the line.

Step 2: Kill the soldiers next to the chosen one (the chosen one will remain alive).

Step 3: Repeat step 1 and step 2 until all the soldiers left belong to the same side (the player or the enemy).

The player wins if all the soldiers left belong to the player. Fall, as the player, wants to know the number of ways to identify the belonging of each soldier so that he can win.

However, soldiers may change their state (the player's troops, the enemies, or unknown). You need to calculate the answers after every change.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \leq T \leq 11)$ – the number of test cases.

For each test case:

In the first line, there are two integer n, q $(1 \leq n, q \leq 2 \times 10^5$, n is **odd**$)$, which are the number of soldiers and the number of operations.

In the second line, there is a string s of length n, which represents the initial soldier state. If s_i is '1', the soldier belongs to you. If s_i is '0', the soldier belongs to your enemy. If s_i is '?', the belonging of this soldier is unknown.

In the next q lines, each contains an integer p $(1 \leq p \leq |s|)$ and a charactor c, which means change s_p to c.

It is guaranteed that the sum of q over all test cases will not exceed 10^6, $s_i, c \in \{\text{'0'}, \text{'1'}, \text{'?'}\}$

Output

For each test case, output the initial answer in a single line. Then output q answers in q lines, which are your answers after each change. All the answers should be output modulo $10^9 + 7$.

Example

standard input	standard output
6	0
5 1	0
01000	0
1 1	1
5 1	247
00000	247
3 1	247
15 4	253
1100????????111	253
8 ?	368
1 0	368
4 1	368
11 ?	736
15 4	1664
0????000?0???0?	3616
6 0	1760
10 0	880
1 ?	0
10 ?	24
15 2	24
???????0?0???0?	
4 0	
1 0	
15 2	
00000000?0???0?	
4 ?	
4 ?	

Problem 1003. Fall with Trees

Input file: `standard input`
Output file: `standard output`

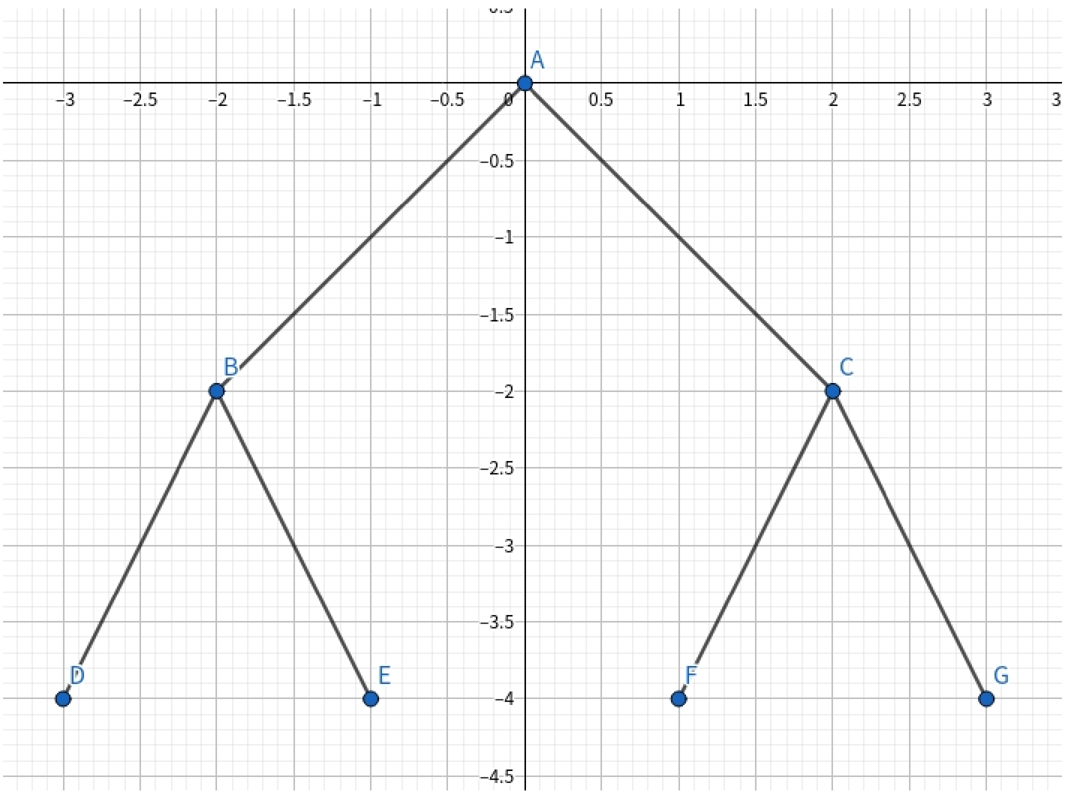
Fall wants to draw a perfect binary tree.

We first stipulate that all nodes in the tree with the same depth also have the same y-coordinate in the plane. Define nodes with the same depth to be nodes at the same level, then the perfect binary tree has four properties.

- It is a full binary tree.
- The difference between the y-coordinates of two nodes at each adjacent level is a constant.
- The difference between the x-coordinates of two adjacent nodes at the same level is constant.
- The x-coordinate of each node is the average of the x-coordinates of its sons.

Fall has drawn the root node and its left and right sons of this binary tree. Now Fall intends to draw a total of k levels and cut the binary tree down and paste it on the wall afterwards, so he wants to know what is the area of the convex hull of all nodes of this perfect binary tree.

Here's the picture of the perfect binary tree for the first example, whose area is $S_{ABC} + S_{BCGD} = 14$



Input

The input consists of multiple test cases.

The first line contains an integer T ($T \leq 2 \times 10^5$) – the number of test cases.

For each test case:

In the first line, there is an integer k $(2 \le k \le 10^4)$.

In the second line, there are six integers $x_{root}, y_{root}, x_{lson}, y_{lson}, x_{rson}, y_{rson} \in [-10^4, 10^4]$,which represent the coordinates of the root node and its sons.

It is guaranteed that all the coordinates meet the conditions of the question, which means:

- $x_{lson} + x_{rson} = 2 \times x_{root}$
- $y_{lson} = y_{rson}$
- $y_{root} > y_{lson}, x_{lson} < x_{rson}$

Output

For each test case, output a real number representing the answer, with three decimal places.

Example

standard input	standard output
3 3 0 0 -2 -2 2 -2 4 0 0 -4 -2 4 -2 10000 0 0 -10000 -10000 10000 -10000	14.000 54.000 3999000000000.000

Problem 1004. Link with Balls

Input file: `standard input`
Output file: `standard output`

There were a lot of balls in the factory of Ball Illusion Technology(BIT). Link, a boy, went there to get some balls, but suddenly, he found that there were too many ways to get balls.

There are $2n$ buckets in the factory. Link may get kx balls from the $2x-1^{th}$ bucket, where k is a non-negtive integer. He may also get at most x balls from the $2x^{th}$ bucket.

Link wanted to get m balls, and he wondered how many ways there were to take out exactly m balls. While Link is calculating the answer, he wants you to calculate it as well, and you should output the answer modulo 10^9+7.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 10^5)$ – the number of test cases.

Each test case contains two integers n and $m(1 \le n, m \le 10^6)$.

Output

For each test case, print the answer modulo 10^9+7 in a single line.

Example

standard input	standard output
4 1 1 2 2 3 3 1000000 1000000	2 6 20 192151600

Problem 1005. Link with EQ

Input file: standard input
Output file: standard output

We often say that details can reflect the level of one's EQ. For example, at BIT, when students study in classrooms (such as A105 in the General Teaching Building) or on long tables in the library, it is often embarrassing and uncomfortable (and occasionally smelly) if another student suddenly sits next to them, which we call low EQ behavior.

Now, there is a long table containing n seats, and the BIT students will choose their seats in the following way in order to avoid low EQ behavior as much as possible.

- If the whole table is empty, randomly choose a position to sit.
- Otherwise, choose a position whose shortest distance to other students is maximized. If there are more than one, randomly choose one of them with equal possibility.

When the farthest distance is 1 (i.e., he will sit next to a classmate anyway), students will sit at a different table to avoid low EQ behavior, and then we will call the table full.

Now, Link wants to know the expected number of people seated in the table when the table is full.

Input

The input consists of multiple test cases.

The first line contains an integer T ($1 \le T \le 10^5$) – the number of test cases.

For each testcase, there is only one integer n ($1 \le n \le 10^6$) in a single line, which is the length of the table.

Output

For each testcase, output the answer modulo $10^9 + 7$.

You may find the way to modulo a rational number in Problem 1006.

Example

standard input	standard output
5 1 2 3 4 2021	1 1 666666673 2 420089906

Problem 1006. Link with Grenade

Input file: `standard input`
Output file: `standard output`

“We are wasting time, let’s go!”

“Terrorists win.”

“Form up, let’s go!”

“Terrorists win.”

......

After being defeated several times, he was so angry that he decided to cheat. He used some tricks to make himself fly really high so that nobody could hurt him. He started to throw grenades, but he suddenly found that he had only 1 hp, which means he would die if the grenade hurt him. Now, Link wants to know the possibility that he will die.

Formally, there is a grenade with initial speed v_0 m/s and explode time t s, which was thrown in a random direction. When the grenade explodes, it will hurt anyone whose distance to it is no more than r. Supposing that the grenade didn’t touch anything before explosion, and the man didn’t move after throwing the grenade, you should output the probability that the man will **survive**.

In order to check your answer precisely, you should output it modulo $10^9 + 7$.

Note1: This is a 3D game. In this game, the acceleration of gravity is 10 m/s^2.

Note2: It can be proved that the answer can always be represented as $\frac{p}{q}$, where the greatest common factor between p and q is 1. When outputting the answer modulo M, you should output such an integer x that $0 \le x < M$ and $x \cdot q \equiv p(mod\ M)$.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 10^5)$ – the number of test cases.

Each test case contains three integers t_0, v_0 and R $(1 \le t_0, v_0, R \le 100)$.

Output

For each test case, print the answer modulo $10^9 + 7$ in a single line.

Example

standard input	standard output
5 1 10 10 1 16 10 1 15 10 1 4 10 1 5 10	625000005 1 1 0 0

Problem 1007. Link with Limit

Input file: `standard input`
Output file: `standard output`

Link has a function $f(x)$, where x and $f(x)$ are both integers in $[1, n]$.

Let $f_n(x) = f(f_{n-1}(x))$ and $f_1(x) = f(x)$, he define the power of a number x as:

$$g(x) = \lim_{n\to+\infty} \frac{1}{n} \sum_{i=1}^{n} f_i(x)$$

He wants to know whether x has the same power for all $x \in [1, n]$.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 100)$ – the number of test cases.

For each test case:

In the first line, there is an integer n $(1 \le n \le 10^5)$.

In the second line, there are n integers, the i-th integer shows the value of $f(i)$ $(1 \le f(i) \le n)$.

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

Output

For each test case, output ‘YES’ if all x have the same power. Otherwise, output ‘NO’.

Example

standard input	standard output
2 2 1 2 2 1 1	NO YES

Problem 1008. Smzzl with Greedy Snake

Input file: `standard input`
Output file: `standard output`

Smzzl is going to make an AI for Greedy Snake. The game goes on xOy plane and there is no obstacles in the plane.

In this game, the snake takes 1 unit of time to move forward for one unit of length. It also takes 1 unit of time for the snake to rotate for 90 degrees. (The snake must rotate for the whole unit of time.) There is a food in the map initially. After the snake eat each food, the next food appears.

Smzzl certainly want the snake to eat the food as fast as possible, so he need to minimize the time when the snake eat each food. Please output a valid operate sequence.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 200)$ – the number of test cases.

For each test case:

In the first line, there are three integers x, y, d $(|x|, |y| \le 10^4,\ 0 \le d \le 3)$. The snake starts on (x, y). d shows the direction of the head of the snake. (0 for y+, 1 for x+, 2 for y-, 3 for x-)

In the second line, there is an integer n $(1 \le n \le 10^5)$, which is the number of foods.

In the next n lines, each contains two integers x, y $(|x|, |y| \le 10^4)$, which means the next food appears at (x, y).

It is guaranteed that **any line that connects two foods that appear adjacently does not parallel to the x-axis or the y-axis**.

Output

For each test case, output the shortest operation sequence. Output ‘f’ for going forward, ‘c’ for rotating clockwise, ‘u’ for rotating counterclockwise. Each operation lasts for one unit of time.

It can be proved that there is only one operation sequence which meets the requirements.

It is guranteed that the total length of output does not exceed 2×10^6.

Example

standard input	standard output
2 0 0 0 2 -1 -1 1 1 0 0 2 5 -1 2 2 4 3 -5 4 -2 5 0	ufufuffuff cfcffffcffffcffffffffufufffffcf

Problem 1009. Smzzl with Safe Zone

Input file: `standard input`
Output file: `standard output`

A boy, whose ID is smzzl, is playing a game named Zig-Zag-Land(ZZL). This is a 2D battle royale game. When the game starts, you will spawn in a random place in the map, and your target is to move into the safe zone.

Given the border of the map and the border of the safe zone, he wants to know the shortest distance between the spawn point and the safe zone in the worst case.

It can be proved that the square of the answer is always rational. In order to check your answer precisely, you need to **print the square of your answer** modulo $10^9 + 7$. You may find the way to modulo a rational number in Problem 1006.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 10^4)$ – the number of test cases.

For each test case:

In the first line, there is an integer m $(3 \le m \le 10^5)$, which is the number of vertexes of the safe zone.

In the next m lines, each line contains two integers x_i, y_i $(|x_i|, |y_i| \le 10^7)$, which is a vertex of the safe zone.

In the next line, there is an integer n $(3 \le n \le 10^5)$, which is the number of vertexes of the game map.

In the next n lines, each line contains two integers x_i, y_i $(|x_i|, |y_i| \le 10^7)$, which is a vertex of the game map.

It is guaranteed that:

- The safe zone is a subset of game map.
- The safe zone and the game map are both convex polygons.
- All vertexes are given in counter-clockwise order.
- The sum of n over all test cases will not exceed 10^6.
- The sum of m over all test cases will not exceed 10^6.

You may need to use faster input method.

Output

For each test case, output the square of your answer modulo $10^9 + 7$.

Example

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

Problem 1010. Smzzl with Tropical Taste

Input file:	`standard input`
Output file:	`standard output`

A boy, whose ID is smzzl, loves drinking Black Ice Tea(BIT), especially the tropical taste one.

The shop owner knows that, and she prepares a swimming pool so that there would be enough space for her to store BIT. When the boy knows that the owner prepared a swimming pool, he starts to drink in the swimming pool. While the owner is pouring BIT into the swimming pool, the boy is drinking it, too.

Assuming that there are V liters of BIT in the swimming pool, the speed the owner pours BIT is qV liters per second, and the speed the boy drinks BIT is pV liters per second. Note that V changes as the time goes.

Now, the owner wants to know whether the following statement is true:

For any $G > 0$, there exists a $T > 0$, for any $t > T$, the boy drinks more than G liters of BIT after t seconds.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 100)$ – the number of test cases.

For each test case, there are two decimals p, q $(0 < p, q < 10^4$, they have at most 4 decimal places) in a single line, p, q are mentioned in the statement. You may regard the initial volume of BIT as 1 liter, and the swimming pool can contain infinity volume of BIT.

Output

For each test case, output a sentence in a single line.

If the statement is true, output 'N0 M0R3 BL4CK 1CE TEA!'.

If the statement is false, output 'ENJ0Y YOURS3LF!'

Example

standard input	standard output
2 1.1 2.2 2.05 1.4	N0 M0R3 BL4CK 1CE TEA! ENJ0Y YOURS3LF!

Problem 1011. Yiwen with Formula

Input file: `standard input`
Output file: `standard output`

Given an array a of length n, for any array b satisfying $1 \le b_i \le n$ and $b_1 < b_2 < \cdots < b_k$ (k is the length of b, $k \ge 1$), calculate:

$$\prod_{b_1<b_2<\cdots<b_k} (a_{b_1} + a_{b_2} + \cdots + a_{b_k})$$

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \le T \le 10)$ – the number of test cases.

For each test case:

The first line contains an integer n $(1 \le n \le 10^5)$, which is the length of a.

The second line contains n integers a_i $(0 \le a_i \le 10^5)$, which is the array a.

It is guaranteed that:

- The sum of n over all test cases will not exceed 2.5×10^5.
- The sum of a_i in a single test will not exceed 10^5.
- The sum of a_i over all test cases will not exceed 4×10^5.

Output

For each test case, output the result modulo 998244353 in a single line.

Example

standard input	standard output
3 2 1 1 3 1 1 2 5 4 6 9 1 5	2 144 417630946

Problem 1012. Yiwen with Sqc

Input file:	`standard input`
Output file:	`standard output`

Recently, Yiwen is learning Assembly Language.

Since Yiwen is a genius in computer science, he proposed an extremely valuable technique called Basic Instruction Technology. Using this kind of technique, Yiwen can create numerous different instruction. Some of these instruction can be used to calculate some special eigenvalue of a string, such as *sqc*.

For a string s, let s_i be the i^{th} character of it.

Instruction sqc has 4 operands s, l, r and c, where s is a string, $1 \leq l \leq r \leq n$, c is a character represented by ASCII value. $sqc(s, l, r, c)$ returns the number of character c in substring $s_l \cdots s_r$.

Yiwen wants you to calculate following expression for the given string s:

$$\sum_{c=97}^{122} \sum_{i=1}^{n} \sum_{j=i}^{n} sqc(s, i, j, c)^2$$

Note that the ASCII value of ‘a’ is 97 and the ASCII value of ‘z’ is 122.

The answer will probably be very huge, so just calculate the answer modulo 998244353.

Input

The input consists of multiple test cases.

The first line contains an integer T $(1 \leq T \leq 100)$ – the number of test cases.

For each test case, the only line contains a string s $(|s| \leq 10^5)$ consists of lowercase English letters.

It is guaranteed that the sum of $|s|$ over all test cases does not excceed 5×10^6.

Output

For each test case, output your answer modulo 998244353 in a single line.

Example

standard input	standard output
5 ababa a abcdefga afidhehfhd ggggggggggg	57 1 122 328 1716