

Problem A. Submissions

Input file:	<i>standard input</i>
Output file:	<i>standard output</i>
Time limit:	2 seconds
Memory limit:	1024 mebibytes

The legend, example, and note of this problem are used fictitiously. Any resemblance to the actual contests, rules, submissions, or teams is coincidental.

In the International Challenging Puzzle Contest (ICPC), there are m submissions. You are given the list of m submissions ordered by time. A submission can be represented as a tuple (c, p, t, s) , which means team c makes a submission on problem p at time t with status s . The status of a submission is either “accepted” or “rejected”.

The score of a team is the pair of the number of problems solved by the team and the total time consumed[†] by the team. The larger the number of problems solved is, the higher the score is. If a tie occurs, the smaller the total time consumed is, the higher the score is.

If team c makes at least one submission with status “accepted” on problem p , we say that team c solves problem p . A team can get a gold medal if the number of teams with higher score is less than $\min(\lceil 0.1 \cdot n \rceil, 35)$, where n is the number of teams that solved **at least** one problem and $\lceil x \rceil$ denotes the smallest integer that is not smaller than x .

You need to find all the teams that can get a gold medal if **at most** one of the m submissions changes its status.

[†] The total time consumed is the sum of times consumed for all solved problems (0 if no problems are solved). The time consumed for a solved problem is the time of the first submission with status “accepted”, plus 20 times the number of submissions on this problem before the first submission with status “accepted”. Note that we say submission i is before submission j if and only if submission i appears earlier than submission j in the given list of m submissions.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^5$) denoting the number of test cases. For each test case:

The first line contains a single integer m ($1 \leq m \leq 10^5$) denoting the number of submissions.

The i -th of the following m lines contains c_i , p_i , t_i , and s_i which mean that team c_i makes a submission on problem p_i at time t_i with status s_i . Specifically:

- c_i is a string of length between 1 and 20 consisting of uppercase letters, lowercase letters, digits and underscores ('_'). Note that no two teams have the same name.
- p_i is an uppercase letter.
- t_i is a non-negative integer less than 300.
- s_i is a string, being either “accepted” or “rejected”.

It is guaranteed that $t_i \leq t_j$ for all $i < j$. Recall that if $t_i = t_j$ and $i < j$, we still say that the i -th submission came before the j -th submission.

It is guaranteed that the sum of m over all test cases does not exceed 10^5 .

Output

For each test case:

Output one integer k on the first line, denoting the number of teams that can get a gold medal if at most one of the m submissions changes its status.

On the second line, output k distinct strings **in any order**, denoting the names of these k teams.

Examples

<i>standard input</i>	<i>standard output</i>
2 5 TSxingxing10 G 0 rejected TSxingxing10 B 83 accepted aoliaoligeiliao J 98 accepted TS1 J 118 accepted TS1 B 263 accepted 12 AllWayTheNorth A 0 rejected YaoYaoLingXian Y 10 accepted XuejunXinyoudui1 X 200 rejected XuejunXinyoudui1 X 200 accepted LetItRot L 215 accepted AllWayTheNorth W 250 accepted ImYourFan I 257 accepted ImYourFan Y 257 accepted AllWayTheNorth T 264 accepted XuejunXinyoudui1 J 294 accepted LetItRot I 299 accepted LetItRot I 299 rejected	2 TSxingxing10 TS1 4 AllWayTheNorth XuejunXinyoudui1 LetItRot ImYourFan

<i>standard input</i>	<i>standard output</i>
2 2 jiangly_fan A 1 accepted jiangly B 23 accepted 3 conqueror_of_tourist A 1 accepted conqueror_of_tourist A 2 accepted tourist B 23 accepted	2 jiangly_fan jiangly 1 conqueror_of_tourist

<i>standard input</i>	<i>standard output</i>
2 13 A A 1 accepted A X 1 accepted K K 1 rejected B B 2 accepted C C 2 accepted D D 2 accepted E E 2 accepted F F 2 accepted G G 2 accepted H H 2 accepted I I 2 accepted J J 2 accepted K K 2 rejected 12 A A 1 accepted A X 1 accepted B B 2 accepted C C 2 accepted D D 2 accepted E E 2 accepted F F 2 accepted G G 2 accepted H H 2 accepted I I 2 accepted J J 2 rejected K K 2 rejected	11 A K B C D E F G H I J 1 A

<i>standard input</i>	<i>standard output</i>
2	2
11	A B
A A 1 accepted	2
B B 1 accepted	A K
C C 2 accepted	
D D 2 accepted	
E E 2 accepted	
F F 2 accepted	
G G 2 accepted	
H H 2 accepted	
I I 2 accepted	
J J 2 accepted	
K K 2 accepted	
12	
A A 1 accepted	
A X 1 accepted	
K K 1 rejected	
B B 2 accepted	
C C 2 accepted	
D D 2 accepted	
E E 2 accepted	
F F 2 accepted	
G G 2 accepted	
H H 2 accepted	
I I 2 accepted	
J J 2 accepted	

Note

In the first case of the first example, **TS1** solves two problems, so they can get a gold medal. **TSxingxing10** can get a gold medal if their first submission changes its status to “accepted”.

In the second case of the first example, **AllWayTheNorth**, **XuejunXinyoudui1**, **LetItRot** and **ImYourFan** have the same score, two problems solved with 514 total time consumed. They can get gold medals simultaneously if no submission changes its status.

Problem B. Festival Decorating

Input file: *standard input*
 Output file: *standard output*
 Time limit: 4 seconds
 Memory limit: 1024 mebibytes

To celebrate the coming winter festival in Byteland, the main street, which can be regarded as the x-axis, is decorated with n colorful lamps, labeled by $1, 2, \dots, n$. The x-coordinate of the i -th lamp is x_i , and the color of the i -th lamp is c_i . No two lamps share the same x-coordinate.

You will be given q queries. In the i -th query, you will be given an integer d_i ($1 \leq d_i \leq 250\,000$), and you need to find the lamp u ($1 \leq u \leq n$) with the minimum index such that there is another lamp located at $x_u + d_i$ and the color of that lamp is different from c_u , or determine it is impossible to find such u . **Your answer is considered correct if its absolute or relative error does not exceed 0.5.**

Input

The first line of the input contains two integers n and q ($1 \leq n, q \leq 250\,000$) denoting the number of lamps and the number of queries.

Each of the next n lines contains two integers x_i and c_i ($1 \leq x_i \leq 250\,000$, $1 \leq c_i \leq n$) denoting the x-coordinate and the color of the i -th lamp. It is guaranteed that no two lamps share the same x-coordinate.

Each of the next q lines contains a single integer d_i ($1 \leq d_i \leq 250\,000$) denoting the i -th query.

Output

For each query, print a line containing a single number: the minimum index u you found. If it is impossible to find such u , print 0 instead.

Your answer is considered correct if its absolute or relative error does not exceed 0.5. Note that this means you can output a non-integer as well.

Formally, let your answer be u , and the jury's answer be u' . Your answer is accepted if and only if:

$$\frac{|u - u'|}{\max(1, |u'|)} \leq 0.5.$$

Example

<i>standard input</i>	<i>standard output</i>
4 5	3
3 1	2
1 2	1
5 1	2
6 2	0
1	
2	
3	
4	
5	

Problem C. Invocations

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 1024 megabytes

In the Polyhedra problem preparation system in the Invocation mode, the solution is run on all the tests available in the problem. For the problem, there is a time limit TL — a number measured in milliseconds.

Let the model solution give a correct answer on each test. For each test, the time T_i during which the solution worked is defined. The time is also measured in a whole number of milliseconds. The results of the run are output in a table, where each cell corresponds to one test, while the following formatting rules are observed:

- If the time taken by the solution for the test is greater than the time limit, the corresponding cell in the table is highlighted in red.
- If the time taken by the solution for the test is less than the time limit, but **strictly** greater than half of the time limit, the corresponding cell in the table is highlighted in orange.
- If none of the cells are highlighted in orange or red, then the cell corresponding to the test on which the maximum time was shown (or several such tests if the maximum time was reached on several tests) is highlighted in blue, provided that the solution worked for **at least** 100 milliseconds.
- All other cells are highlighted in green.

Given the number of tests, the value of TL , and the result of the solution on the tests, output the colors corresponding to the tests in the Polyhedra system run results table.

Input

The first line of the input contains a single integer N ($1 \leq N \leq 1000$) — the number of tests on which the model solution is run.

The second line contains a single integer TL ($1 \leq TL \leq 15\,000$) — the time limit for the solution to run on one test.

Each of the next N lines contains a single integer T_i ($1 \leq T_i \leq 20\,000$) — the time the solution worked on the i -th test.

Output

Output N lines. If the cell corresponding to the i -th test is highlighted in red, output “**red**” in the i -th line. If the cell is highlighted in orange, output “**orange**”. If the cell is highlighted in blue, output “**blue**”, and if the cell is highlighted in green, output “**green**”.

Examples

standard input	standard output
5 1000 10 100 85 500 499	green green green blue green
5 1000 1001 501 1000 500 500	red orange orange green green
1 1000 42	green

Problem D. Operator Precedence

Input file: *standard input*
 Output file: *standard output*
 Time limit: 1 second
 Memory limit: 1024 mebibytes

Randias is facing his primary school homework:

Find a nonzero integer sequence a of length $2n$ satisfying

$$(a_1 \times a_2) + (a_3 \times a_4) + \dots + (a_{2n-1} \times a_{2n}) \\ = a_1 \times (a_2 + a_3) \times (a_4 + a_5) \times \dots \times (a_{2n-2} + a_{2n-1}) \times a_{2n} \neq 0.$$

In shorter form, $\sum_{i=1}^n a_{2i-1} a_{2i} = a_1 a_{2n} \prod_{i=2}^n (a_{2i-2} + a_{2i-1}) \neq 0$.

Of course, Randias knows how to solve it. But he wants to give you a test. Can you solve the question above?

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^5$) denoting the number of test cases.

For each test case, the only line contains a single integer n ($2 \leq n \leq 10^5$).

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output one line with $2n$ integers: a_1, a_2, \dots, a_{2n} ($1 \leq |a_i| \leq 10^{10}$).

It can be shown that the answer always exists.

If there are several possible answers, output any one of them.

Example

<i>standard input</i>	<i>standard output</i>
3	1 -3 -3 1
2	1 -10 6 6 -10 1
3	1 -15 10 -1 -1 10 -15 1
4	

Problem E. Period of a String

Input file: *standard input*
 Output file: *standard output*
 Time limit: 1 second
 Memory limit: 1024 mebibytes

Randias has n strings s_1, s_2, \dots, s_n .

For two strings $a = \overline{a_0 a_1 \dots a_{p-1}}$ and $b = \overline{b_0 b_1 \dots b_{q-1}}$, if for all i ($0 \leq i < q$), $b_i = a_{i \bmod p}$, we say that a is a period of b .

Now, Randias can perform the following operation:

- Choose one string s_i and choose two indices j and k ($0 \leq j, k < |s_i|$), then swap $s_{i,j}$ and $s_{i,k}$.

He can perform this operation **any** number of times. After all the operations, he wants the following to be true: for each $1 < i \leq n$, string s_{i-1} is a period of s_i .

Help him to find the possible final strings, or determine it is impossible.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^4$) denoting the number of test cases. For each test case:

The first line contains a single integer n ($2 \leq n \leq 10^5$).

Then follow n lines. The i -th of these lines contains the string s_i ($1 \leq |s_i| \leq 5 \cdot 10^6$). It is guaranteed that the strings only contain lowercase English letters.

It is guaranteed that the sum of n does not exceed 10^5 , and the sum of $|s_i|$ does not exceed $5 \cdot 10^6$.

Output

For each test case, if it is possible to make s_{i-1} a period of s_i for all i after some operations, output “YES” (without quotes) on the first line. Then output n strings in n lines. The i -th string s'_i represents the i -th string after all operations. If there are multiple answers, output any one of them.

If it is impossible to do that, output “NO” (without quotes) on the first line.

Example

<i>standard input</i>	<i>standard output</i>
4	NO
2	YES
abc	abbca
abcd	abbc
4	abbcabb
bbcaa	a
cabb	YES
acabbbb	ab
a	aba
3	abaabaab
ab	NO
aab	
bbaaaaab	
3	
ab	
aab	
bbaaaaaa	

Problem F. Top Cluster

Input file: *standard input*
 Output file: *standard output*
 Time limit: 2 seconds
 Memory limit: 1024 mebibytes

Top Cluster is a useful data structure for maintaining data on a tree. Using Top Cluster, we can do range queries efficiently.

Lovely EMmm likes data structure technologies very much. She is learning Top Cluster now, and she is trying to solve a data structure problem. Can you write a program to solve the problem together with EMmm?

In the problem, you will be given a tree with n vertices, labeled by $1, 2, \dots, n$. The value of the i -th vertex is a non-negative integer w_i . **All the values are pairwise distinct.**

You will then be given q queries. In the i -th query, you will be given two integers x_i and k_i ($1 \leq x_i \leq n$, $0 \leq k_i \leq 10^{15}$), and you need to find the value of $\text{mex}(\{w_u \mid \text{dist}(u, x_i) \leq k_i \wedge 1 \leq u \leq n\})$.

Here, $\text{dist}(u, v)$ denotes the length of the shortest path from vertex u to vertex v . In mathematics, the **mex** (“**m**inimum **e**xcluded value”) of a set is the smallest non-negative integer that does not belong to the set.

EMmm is good at solving mex problems. She found that when all the values are pairwise distinct, the problem above is equivalent to finding the smallest non-negative integer that either occurred outside the given range, which means $\text{dist}(x_i, u) > k_i$, or never occurred in the whole tree. However, she can't go any further. Can you help her solve the problem?

Input

The first line of the input contains two integers n and q ($1 \leq n, q \leq 5 \cdot 10^5$) denoting the number of vertices and the number of queries.

The second line contains n integers w_1, w_2, \dots, w_n ($0 \leq w_i \leq 10^9$) denoting the values of the vertices. **It is guaranteed that all the values are pairwise distinct.**

Each of the next $n - 1$ lines contains three integers u_i, v_i and ℓ_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$, $1 \leq \ell_i \leq 10^9$) denoting a two-way edge between vertices u_i and v_i with length ℓ_i . It is guaranteed that the input forms a tree.

Each of the next q lines contains two integers x_i and k_i ($1 \leq x_i \leq n$, $0 \leq k_i \leq 10^{15}$) denoting the i -th query.

Output

For each query, print a single line containing an integer: the mex value that you found.

Example

<i>standard input</i>	<i>standard output</i>
5 4	1
3 9 0 1 2	0
1 2 10	3
3 1 4	4
3 4 3	
3 5 2	
3 0	
1 0	
4 6	
4 7	

Problem G. Snake Move

Input file: *standard input*
 Output file: *standard output*
 Time limit: 2 seconds
 Memory limit: 1024 mebibytes

Putata is playing a famous snake game on his laptop, where a snake moves around on a grid of size $n \times m$. There may be obstacles in some cells of the grid. The snake can be represented as a sequence of coordinate pairs that determine where its body is located: $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$. Here, k denotes the length of the snake. The head of the snake is at (x_1, y_1) , the tail is at (x_k, y_k) , and neighboring parts of the body are located in cells that share a side.

In this game, the snake is programmed with a series of commands represented as a string. There are 5 types of commands that you can use:

- ‘L’: Command the snake to move one step left. The head will then move to $(x_1, y_1 - 1)$.
- ‘R’: Command the snake to move one step right. The head will then move to $(x_1, y_1 + 1)$.
- ‘U’: Command the snake to move one step up. The head will then move to $(x_1 - 1, y_1)$.
- ‘D’: Command the snake to move one step down. The head will then move to $(x_1 + 1, y_1)$.
- ‘S’: Shorten the length of the snake by one. The tail of the body will be erased. The length will become $k - 1$. Note that you can not do this when $k = 1$.

When the head moves, each part of the body also moves accordingly. Specifically, the i -th part of the body ($2 \leq i \leq k$) moves to the position where the $(i - 1)$ -st part was before the command. The snake can not move into a cell with an obstacle, and can not move outside the grid. Besides, the snake cannot collide with itself. So you should guarantee that no two parts of the body will share the same location.

Consider the following corner case: The head is at (x_1, y_1) , and the tail is at (x_k, y_k) . If the head is moving to (x'_1, y'_1) , then it is **allowed** to move to $(x'_1, y'_1) = (x_k, y_k)$: if we think about a real-world scenario, the head moves into the cell just as the tail moves out. In a similar fashion, it is **allowed** to swap the head and the tail by using a single command when $k = 2$.

You will be given the map of the grid and the body sequence of the snake. Let $f(i, j)$ denote the minimum number of commands that Putata needs to use such that the head of the snake will arrive at (i, j) , or 0 if it is impossible. You have to calculate:

$$\left(\sum_{i=1}^n \sum_{j=1}^m f(i, j)^2 \right) \bmod 2^{64}.$$

Input

The first line of the input contains three integers n , m and k ($1 \leq n, m \leq 3000$, $1 \leq k \leq \min\{nm, 10^5\}$) denoting the size of the grid and the length of the snake.

In the next k lines, the i -th line contains two integers x_i and y_i ($1 \leq x_i \leq n$, $1 \leq y_i \leq m$, $|x_i - x_{i+1}| + |y_i - y_{i+1}| = 1$) denoting the location of the i -th part of the body. It is guaranteed that all the k pairs (x_i, y_i) are pairwise distinct. It is also guaranteed that each part is in a cell without an obstacle.

In the next n lines, the i -th line contains a string of length m . If cell (i, j) is empty, the j -th character in the i -th of these lines is ‘.’. If cell (i, j) is occupied by an obstacle, the character is ‘#’.

Output

Print a single line containing an integer: the answer to the problem.

Examples

<i>standard input</i>	<i>standard output</i>
4 5 5 3 5 3 4 3 3 3 2 4 2	293
2 2 4 1 1 1 2 2 2 2 1	14
5 5 3 1 2 1 1 2 1###. .#.#. .###.	407

Problem H. Sugar Sweet II

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 1024 mebibytes

Sugar is sweet.

There are n children asking for sugar. Prof. Chen gives out sugar to the children. The i -th child initially has a_i bags of sugar. There are n events happening in **uniformly randomized order**. The i -th event is:

- If the i -th child has **strictly less** bags of sugar than the b_i -th child, then the i -th child will get extra w_i bags of sugar. Otherwise, nothing happens.

Now, since the events happen in random order, Randias, which is the assistant of Prof. Chen, wants to know the expected number of bags of sugar each child will have after all the events happen.

It can be shown that the answer can be expressed as an irreducible fraction $\frac{x}{y}$ where x and y are integers and $y \not\equiv 0 \pmod{10^9 + 7}$. Output the integer equal to $x \cdot y^{-1} \pmod{10^9 + 7}$. In other words, output such an integer a that $0 \leq a < 10^9 + 7$ and $a \cdot y \equiv x \pmod{10^9 + 7}$.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 5 \cdot 10^5$) denoting the number of test cases. For each test case:

The first line contains a single integer n ($1 \leq n \leq 5 \cdot 10^5$) denoting the number of children.

The second line contains n integers a_i ($1 \leq a_i \leq 10^9$): the initial number of bags of sugar each child has.

The third line contains n integers b_i ($1 \leq b_i \leq n$).

The fourth line contains n integers w_i ($1 \leq w_i \leq 10^9$).

It is guaranteed that the sum of n over all test cases does not exceed $5 \cdot 10^5$.

Output

For each test case, output n integers in a line: the expected number of bags of sugar each child will get. Output the answers as integers modulo $10^9 + 7$, as described above.

Example

<i>standard input</i>	<i>standard output</i>
4	500000007 5 5 6
4	5 10 9
2 5 5 2	166666673 5 6
4 2 1 3	500000006 4 3 4 5
3 2 1 4	
3	
5 4 3	
1 1 1	
6 6 6	
3	
5 4 3	
2 3 1	
1 2 3	
5	
2 1 3 2 1	
5 1 1 3 4	
1 3 4 2 4	

Problem I. Exceptional Set

Input file: **standard input**
 Output file: **standard output**
 Time limit: 1 second
 Memory limit: 1024 megabytes

Given a string s consisting of lowercase Latin letters. A set of n **non-empty** strings a_i is called *exceptional* if:

- The concatenation of the strings a_i , that is, the string obtained by writing the strings from a_1 to a_n (in increasing index order) one after the other closely from left to right, would coincide with s .
- All strings a_i would be palindromes (that is, they would coincide with themselves when read from right to left).
- n , i.e. the number of these strings, would be as large as possible.

Output any of the exceptional sets for the given string.

Input

The input contains a single string s , consisting of at least one and at most 1000 lowercase English letters.

Output

In the first line, output a single number — the number of strings n . Then output the strings a_i , separated by a line break, in increasing index order.

If it is impossible to split the string in this way, output -1 . If there are multiple exceptional sets, output any.

Examples

standard input	standard output
ab	2 a b
c	1 c

Problem J. Mysterious Tree

Input file: *standard input*
 Output file: *standard output*
 Time limit: 1 second
 Memory limit: 1024 mebibytes

This is an interactive problem.

Randias has an unknown hidden tree with n vertices. The tree is either a **chain** or a **star**. Randias now needs to determine whether the tree is a chain or a star. He can ask a question in the following form, but no more than $\lceil \frac{n}{2} \rceil + 3$ times:

- Is there an edge between vertex u and vertex v ($1 \leq u, v \leq n, u \neq v$)?

Randias needs to determine which of the two kinds the tree is. Help him to ask the questions and determine the answer.

A tree is called a **chain** if and only if there exists a permutation p_1, p_2, \dots, p_n such that, for every i ($1 \leq i < n$), there is an edge (p_i, p_{i+1}) in the tree. Here, a permutation of length n is an array where each integer from 1 to n appears exactly once.

A tree is called a **star** if and only if there exists a vertex u such that, for every other vertex v , there is an edge (u, v) in the tree.

In this problem, the interactor is **adaptive**, which means that the secret tree is not fixed beforehand. Instead, the interactor can change the tree arbitrarily during the interaction. Nevertheless, at every moment, the tree will be consistent with all the answers you got.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 250$) denoting the number of test cases.

For each test case, the first line contains one integer n ($4 \leq n \leq 1000$) denoting the number of vertices. It is guaranteed that the sum of n over all test cases does not exceed 1000.

Interaction Protocol

You can ask at most $\lceil \frac{n}{2} \rceil + 3$ questions in every test case. To ask a question, output a line of the form “? u v ” ($1 \leq u, v \leq n, u \neq v$). Then you should read the response from standard input.

In response to the query, the interactor will output a line with a single integer: 1 if there is an edge between u and v in the tree, or 0 if there is no such edge.

To give your answer, print a line of the form “! 1” if you determined that the tree is a chain, or “! 2” if you determined that it is a star. The output of the answer is **not** counted towards the limit of $\lceil \frac{n}{2} \rceil + 3$ queries.

After printing the answer, your program should process the next test case, or terminate if there are no more test cases.

After printing each line, do not forget to output end of line and flush the output. To do the latter, you can use `fflush(stdout)` or `cout.flush()` in C++, `System.out.flush()` in Java, or `stdout.flush()` in Python.

Example

<i>standard input</i>	<i>standard output</i>
2	
4	
	? 1 2
1	
	? 2 3
1	
	? 3 4
1	
	! 1
4	
	? 1 3
1	
	? 2 4
0	
	? 1 2
0	
	? 1 4
0	
	! 2

Problem K. Diversity Of Strings

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

We will call a string *basic* if the string consists of identical letters.

We define the *diversity* of a string consisting of lowercase letters as the minimum number of basic strings that the string can be cut into. For example, the diversity of the string `aaaa` is 1, the diversity of the string `aabb` is 2 (the strings `aa` and `bb`), and the diversity of the string `abc` is 3 (the strings `a`, `b`, and `c`).

You are given a template — a string consisting of lowercase Latin letters and the symbol `*`. You need to find the number of the most uniform strings (i.e., strings with the minimum possible diversity) that can be obtained by replacing each asterisk with any lowercase letter.

As the answer can be very large, output the remainder of its division by 998 244 353.

Input

The input contains a non-empty string consisting of lowercase English letters and the symbol `*`. The length of the string does not exceed 10^4 .

Output

Output a single number — the remainder of the division by 998 244 353 of the number of the most uniform strings among all the strings obtained by replacing each asterisk with any English letter.

Examples

standard input	standard output
<code>f**s***p**</code>	12
<code>contest</code>	1

Problem L. Champoins League

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

After the next reform, matches of the Champions League will be played in groups of N teams. The teams play each other twice (one match on the field of one team, the other on the field of the other team). A win in a match gives 3 points, a tie gives 1, and a loss gives 0.

The teams have played both rounds, scoring a total of K points. Is this possible, and if so, how many matches ended in a tie?

Input

The first line of the input contains a single integer N ($4 \leq N \leq 100$) — the number of teams in the group. The second line contains a single integer K ($0 \leq K \leq 10^5$) — the total points scored by all teams in a full two-round tournament.

Output

Output -1 if there are no match outcomes where the total points scored by all teams is equal to K . Otherwise, output a single integer — the number of matches that are tied.

Examples

standard input	standard output
4 26	10
4 37	-1

Problem M. V-Diagram

Input file: *standard input*
 Output file: *standard output*
 Time limit: 1 second
 Memory limit: 1024 mebibytes

A 1-indexed integer sequence a of length n is a V-diagram if $n \geq 3$ and there exists an index i ($1 < i < n$) satisfying the following:

- $a_j > a_{j+1}$ for $1 \leq j < i$;
- $a_j > a_{j-1}$ for $i < j \leq n$.

Given a V-diagram a , find a V-diagram b with the maximum possible average such that b is a consecutive subsequence of a .

A consecutive subsequence of a sequence can be obtained by removing some (possibly zero) elements from the beginning and end of the sequence.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^5$) denoting the number of test cases. For each test case:

The first line contains one integer n ($3 \leq n \leq 3 \cdot 10^5$) denoting the length of the integer sequence a .

The second line contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) denoting the sequence a itself.

It is guaranteed that a is a V-diagram, and the sum of n over all test cases does not exceed $3 \cdot 10^5$.

Output

For each test case, output a real number denoting the maximum possible average.

Your answer is considered correct if its absolute or relative error does not exceed 10^{-9} .

Formally, let your answer be x , and the jury's answer be y . Your answer will be considered correct if and only if $\frac{|x-y|}{\max(1,|y|)} \leq 10^{-9}$.

Example

<i>standard input</i>	<i>standard output</i>
2	6.75000000000000000000000000000000
4	5.83333333333333333333333333333333
8 2 7 10	
6	
9 6 5 3 4 8	