The 16th Zhejiang Provincial Collegiate Programming Contest

Sponsored by

图森 tuSimple

Contest Session

April 27, 2019

Problem List

A	Vertices in the Pocket
B	Element Swapping
C	Array in the Pocket
D	Traveler
E	Sequence in the Pocket
F	Abbreviation
G	Lucky 7 in the Pocket
H	Singing Everywhere
I	Fibonacci in the
J	Pocket Welcome Party
K	Strings in the Pocket
L	Square on the Plane
M	Trees in the Pocket

This problem set should contain 13 (thirteen) problems on 13 (thirteen) numbered pages. Please inform a runner immediately if something is missing from your problem set.

Prepared by SUA Programming Contest Problem Setter Team.
https://sua.ac/

Hosted by

Problem Set Prepared by

Problem A. Vertices in the Pocket

DreamGrid has just found an undirected simple graph with n vertices and no edges (that's to say, it's a graph with n isolated vertices) in his right pocket, where the vertices are numbered from 1 to n. Now he would like to perform q operations of the following two types on the graph:

- 1 a b – Connect the a-th vertex and the b-th vertex by an edge. It's guaranteed that before this operation, there does not exist an edge which connects vertex a and b directly.
- 2 k – Find the answer for the query: What's the minimum and maximum possible number of connected components after adding k new edges to the graph. Note that after adding the k edges, the graph must still be a simple graph, and the query does NOT modify the graph.

Please help DreamGrid find the answer for each operation of the second type. Recall that a simple graph is a graph with no self loops or multiple edges.

Input

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

The first line contains two integers n and q ($1 \le n \le 10^5$, $1 \le q \le 2 \times 10^5$), indicating the number of vertices and the number of operations.

For the following q lines, the i-th line first contains an integer p_i ($p_i \in \{1, 2\}$), indicating the type of the i-th operation.

- If $p_i = 1$, two integers a_i and b_i follow ($1 \le a_i, b_i \le n$, $a_i \ne b_i$), indicating an operation of the first type. It's guaranteed that before this operation, there does not exist an edge which connects vertex a and b directly.
- If $p_i = 2$, one integer k_i follows ($0 \le k_i \le \frac{n(n-1)}{2}$), indicating an operation of the second type. It's guaranteed that after adding k_i edges to the graph, the graph is still possible to be a simple graph.

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

Output

For each operation of the second type output one line containing two integers separated by a space, indicating the minimum and maximum possible number of connected components in this query.

Example

standard input	standard output
1 5 5 1 1 2 2 1 1 1 3 2 1 2 3	3 3 2 3 1 2

Problem B. Element Swapping

DreamGrid has an integer sequence $a_1, a_2, \dots, a_n$ and he likes it very much. Unfortunately, his naughty roommate BaoBao swapped two elements a_i and a_j ($1 \le i < j \le n$) in the sequence when DreamGrid wasn't at home. When DreamGrid comes back, he finds with dismay that his precious sequence has been changed into $a_1, a_2, \dots a_{i-1}, a_j, a_{i+1}, \dots, a_{j-1}, a_i, a_{j+1}, \dots, a_n$!

What's worse is that DreamGrid cannot remember his precious sequence. What he only remembers are the two values

$$x = \sum_{k=1}^{n} k a_k \qquad \text{and} \qquad y = \sum_{k=1}^{n} k a_k^2$$

Given the sequence after swapping and the two values DreamGrid remembers, please help DreamGrid count the number of possible element pairs (a_i, a_j) BaoBao swaps.

Note that as DreamGrid is poor at memorizing numbers, the value of x or y might not match the sequence, and no possible element pair can be found in this situation.

Two element pairs (a_i, a_j) $(1 \le i < j \le n)$ and (a_p, a_q) $(1 \le p < q \le n)$ are considered different if $i \ne p$ or $j \ne q$.

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 three integers n, x and y ($2 \le n \le 10^5, 1 \le x, y \le 10^{18}$), indicating the length of the sequence and the two values DreamGrid remembers.

The second line contains n integers $b_1, b_2, \dots, b_n$ ($1 \le b_i \le 10^5$), indicating the sequence after swapping. It's guaranteed that $\sum\limits_{k=1}^{n} k b_k \le 10^{18}$ and $\sum\limits_{k=1}^{n} k b_k^2 \le 10^{18}$.

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 number of possible element pairs BaoBao swaps.

Example

standard input	standard output
2 6 61 237 1 1 4 5 1 4 3 20190429 92409102 1 2 3	2 0

Note

For the first sample test case, it's possible that BaoBao swaps the 2nd and the 3rd element, or the 5th and the 6th element.

Problem C. Array in the Pocket

BaoBao has just found an array $A = \{a_1, a_2, \dots, a_n\}$ of n integers in his left pocket. As BaoBao is bored, he decides to rearrange it into another array $B = \{b_1, b_2, \dots, b_n\}$ of n integers such that B is a permutation of A, and $a_i \neq b_i$ for all $1 \le i \le n$. Please help BaoBao finish the rearragement.

If multiple valid rearrangements exist, find the smallest one among them.

Consider two arrays $C = \{c_1, c_2, \dots, c_n\}$ and $D = \{d_1, d_2, \dots, d_n\}$, we say C is smaller than D if there exists an integer k such that $1 \le k \le n$, $c_i = d_i$ for all $1 \le i < k$, and $c_k < d_k$.

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 ($1 \le n \le 10^5$), indicating the length of the array.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \le a_i \le n$), indicating the given array.

Output

For each test case output one line containing n integers $b_1, b_2, \dots, b_n$ separated by a space, indicating the answer. If there are multiple valid answers, output the smallest one. If no valid answer exists, print "Impossible" (without quotes) instead.

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

Example

standard input	standard output
3 4 4 1 3 2 4 1 1 2 3 3 1 1 1	1 2 4 3 2 3 1 1 Impossible

Problem D. Traveler

The famous traveler BaoBao is visiting the Dream Kingdom now. There are n cities in Dream Kingdom, numbered from 1 to n. The cities are connected by **directed** roads. For all $1 \le i \le n$:

- There is a road from the i-th city to the $(i-1)$-th city if $1 \le i-1 \le n$.
- There is a road from the i-th city to the $2i$-th city if $1 \le 2i \le n$.
- There is a road from the i-th city to the $(2i+1)$-th city if $1 \le 2i+1 \le n$.
- There is a road from the i-th city to the $\lfloor \frac{i}{2} \rfloor$-th city if $1 \le \lfloor \frac{i}{2} \rfloor \le n$, where $\lfloor \frac{i}{2} \rfloor$ indicates the largest integer x such that $2x \le i$.

BaoBao starts his travel from the 1st city. As he doesn't like visiting a city more than once, he wants to find a route which goes through each of the n cities exactly once. Can you help him find such a route?

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 the only line contains an integer n ($1 \le n \le 10^5$), indicating the number of cities in Dream Kingdom.

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. If there exists a route which starts from the 1st city and visits each city exactly once, output n integers $c_1, c_2, \dots, c_n$ separated by a space, where c_i indicates the i-th city in the route (note that according to the description, there must be $c_1 = 1$). If there is no valid route, output "-1" (without quotes) instead. If there are multiple valid answers, you can output 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
2 2 9	1 2 1 3 6 5 2 4 9 8 7

Problem E. Sequence in the Pocket

DreamGrid has just found an integer sequence $a_1, a_2, \ldots, a_n$ in his right pocket. As DreamGrid is bored, he decides to play with the sequence. He can perform the following operation any number of times (including zero time): select an element and move it to the beginning of the sequence.

What's the minimum number of operations needed to make the sequence non-decreasing?

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 ($1 \le n \le 10^5$), indicating the length of the sequence.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$), indicating the given sequence.

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 answer.

Example

standard input	standard output
2 4 1 3 2 4 5 2 3 3 5 5	2 0

Note

For the first sample test case, move the 3rd element to the front (so the sequence become $\{2, 1, 3, 4\}$), then move the 2nd element to the front (so the sequence become $\{1, 2, 3, 4\}$). Now the sequence is non-decreasing.

For the second sample test case, as the sequence is already sorted, no operation is needed.

Problem F. Abbreviation

In the *Test of English as a Foreign Language* (TOEFL), the listening part is very important but also very hard for most students since it is usually quite hard for them to remember the whole passage. To help themselves memorize the content, students can write down some necessary details. However, it is not easy to write down the complete word because of its length. That's why we decide to use the abbreviation to express the whole word.

It is very easy to get the abbreviation, all we have to do is to keep the consonant letters and erase the vowel. In the English alphabet, we regard 'a', 'e', 'i', 'y', 'o', 'u' as the vowels and the other letters as the consonants. For example, "subconscious" will be expressed as "sbcnscs".

However, there is one exception: if the vowel appears as the first letter, they should be kept instead of thrown away. For example, "oipotato" should be expressed as "optt".

Since you have learned how to use the abbreviation method, it's time for some exercises. We now present you T words in total, it's your task to express them in their abbreviation form.

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 only line contains a string s ($1 \le |s| \le 100$) consisting of lowercase English letters, indicating the word which needs to be abbreviated.

Output

For each test case output one line containing one string, which is the correct abbreviation of the given word.

Example

standard input	standard output
5 subconscious oipotato word symbol apple	sbcnscs optt wrd smbl appl

Problem G. Lucky 7 in the Pocket

BaoBao loves number 7 but hates number 4, so he refers to an integer x as a "lucky integer" if x is divisible by 7 but not divisible by 4. For example, 7, 14 and 21 are lucky integers, but 1, 4 and 28 are not.

Today BaoBao has just found an integer n in his left pocket. As BaoBao dislikes large integers, he decides to find a lucky integer m such that $m \geq n$ and m is as small as possible. Please help BaoBao calculate the value of m.

Input

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

The first and only line contains an integer n ($1 \leq n \leq 100$), indicating the integer in BaoBao's left pocket.

Output

For each test case output one line containing one integer, indicating the value of m.

Example

standard input	standard output
4 1 7 20 28	7 7 21 35

Problem H. Singing Everywhere

Baobao loves singing very much and he really enjoys a game called *Singing Everywhere*, which allows players to sing and scores the players according to their performance.

Consider the song performed by Baobao as an integer sequence $a_1, a_2, \dots, a_n$, where a_i indicates the i-th note in the song. We say a note a_k is a "voice crack" if $1 < k < n$, $a_{k-1} < a_k$ and $a_{k+1} < a_k$. The more voice cracks BaoBao sings, the lower score he gets.

To get a higher score, BaoBao decides to delete at most one note in the song. What's the minimum number of times BaoBao sings a voice crack after this operation?

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 one integer n ($1 \le n \le 10^5$), indicating the length of the song.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($-2^{31} \le a_i < 2^{31}$), indicating the song performed by BaoBao.

It's guaranteed that at most 5 test cases have $n > 100$.

Output

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

Example

standard input	standard output
3 6 1 1 4 5 1 4 7 1 9 1 9 8 1 0 10 2 1 4 7 4 8 3 6 4 7	1 0 2

Note

For the first sample test case, BaoBao does not need to delete a note. Because if he deletes no note, he will sing 1 voice crack (the 4th note), and no matter which note he deletes, he will also always sing 1 voice crack.

For the second sample test case, BaoBao can delete the 3rd note, and no voice cracks will be performed. Yay!

For the third sample test case, BaoBao can delete the 4th note, so that only 2 voice cracks will be performed (4 **8** 3 and 3 **6** 4).

Problem I. Fibonacci in the Pocket

DreamGrid has just found a Fibonacci sequence $f_1, f_2, \ldots$ and two integers a and b in his right pocket, where f_k indicates the k-th element in the Fibonacci sequence.

Please tell DreamGrid if $\sum_{i=a}^{b} f_i$ is even or is odd.

Recall that a Fibonacci sequence is an infinite sequence which satisfies $f_1 = 1$, $f_2 = 1$ and $f_i = f_{i-1} + f_{i-2}$ for all $i \geq 3$.

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 and only line contains two integers a and b ($1 \leq a \leq b < 10^{10000}$). Their meanings are described above.

Output

For each test case output one line. If $\sum_{i=a}^{b} f_i$ is even output "0" (without quotes); If $\sum_{i=a}^{b} f_i$ is odd output "1" (without quotes).

Example

standard input	standard output
6 1 2 1 3 1 4 1 5 123456 12345678987654321 123 20190427201904272019042720190427	0 0 1 0 0 1

Note

The first few elements of the Fibonacci sequence are: $f_1 = 1$, $f_2 = 1$, $f_3 = 2$, $f_4 = 3$, $f_5 = 5$, $f_6 = 8$...

Problem J. Welcome Party

The 44th World Finals of the International Collegiate Programming Contest (ICPC 2020) will be held in Moscow, Russia. To celebrate this annual event for the best competitive programmers around the world, it is decided to host a welcome party for all n participants of the World Finals, numbered from 1 to n for convenience.

The party will be held in a large hall. For security reasons, all participants must present their badge to the staff and pass a security check in order to be admitted into the hall. Due to the lack of equipment to perform the security check, it is decided to open only one entrance to the hall, and therefore only one person can enter the hall at a time.

Some participants are friends with each other. There are m pairs of mutual friendship relations. Needless to say, parties are more fun with friends. When a participant enters the hall, if he or she finds that none of his or her friends is in the hall, then that participant will be unhappy, even if his or her friends will be in the hall later. So, one big problem for the organizer is the order according to which participants enter the hall, as this will determine the number of unhappy participants. You are asked to find an order that minimizes the number of unhappy participants. Because participants with smaller numbers are more important (for example the ICPC director may get the number 1), if there are multiple such orders, you need to find the lexicographically smallest one, so that important participants enter the hall first.

Please note that if participant a and b are friends, and if participant b and c are friends, it's NOT necessary that participant a and c are friends.

Input

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

The first line contains two integers n and m ($1 \le n, m \le 10^6$), the number of participants and the number of friendship relations.

The following m lines each contains two integers a and b ($1 \le a, b \le n, a \ne b$), indicating that the a-th and the b-th participant are friends. Each friendship pair is only described once in the input.

It is guaranteed that neither the sum of n nor the sum of m of all cases will exceed 10^6.

Output

For each case, print a single integer on the first line, indicating the minimum number of unhappy participants. On the second line, print a permutation of 1 to n separated by a space, indicating the lexicographically smallest ordering of participants entering the hall that achieves this minimum number.

Consider two orderings $P = p_1, p_2, \ldots, p_n$ and $Q = q_1, q_2, \ldots, q_n$, we say P is lexicographically smaller than Q, if there exists an integer k ($1 \le k \le n$), such that $p_i = q_i$ holds for all $1 \le i < k$, and $p_k < q_k$.

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

Example

standard input	standard output
2 4 3 1 2 1 3 1 4 4 2 1 2 3 4	1 1 2 3 4 2 1 2 3 4

Problem K. Strings in the Pocket

BaoBao has just found two strings $s = s_1s_2 \dots s_n$ and $t = t_1t_2 \dots t_n$ in his left pocket, where s_i indicates the i-th character in string s, and t_i indicates the i-th character in string t.

As BaoBao is bored, he decides to select a substring of s and reverse it. Formally speaking, he can select two integers l and r such that $1 \le l \le r \le n$ and change the string to $s_1s_2 \dots s_{l-1}s_rs_{r-1} \dots s_{l+1}s_ls_{r+1} \dots s_{n-1}s_n$.

In how many ways can BaoBao change s to t using the above operation exactly once? Let (a, b) be an operation which reverses the substring $s_as_{a+1} \dots s_b$, and (c, d) be an operation which reverses the substring $s_cs_{c+1} \dots s_d$. These two operations are considered different, if $a \ne c$ or $b \ne d$.

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 string s ($1 \le |s| \le 2 \times 10^6$), while the second line contains another string t ($|t| = |s|$). Both strings are composed of lower-cased English letters.

It's guaranteed that the sum of $|s|$ of all test cases will not exceed 2×10^7.

Output

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

Example

standard input	standard output
2 abcbcdcbd abcdcbcbd abc abc	3 3

Note

For the first sample test case, BaoBao can do one of the following three operations: (2, 8), (3, 7) or (4, 6).

For the second sample test case, BaoBao can do one of the following three operations: (1, 1), (2, 2) or (3, 3).

Problem L. Square on the Plane

Consider a point $P(x_p, y_p)$ on the two-dimensional plane. Starting from $(a, 0)$, point P moves along the circumference of the ellipse $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ with a uniform speed of 1 unit per second in counter-clockwise direction.

A moving square S, whose sides are parallel with the x-axis or the y-axis, takes P as its center and changes its side length during its movement. Precisely speaking, the side length of S is exactly $2|y_p|$.

What's the total area passed through by S after point P moving for t seconds?

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 real numbers a, b, t ($1 \le \frac{a}{2} \le b \le a \le 100, 1 \le t \le 1000$) with at most six digits after the decimal point, indicating the length of the semi-major axis of the ellipse, the length of the semi-minor axis of the ellipse and the moving time in seconds.

Output

For each test case output one line, indicating the total area passed through by the moving square. Your answer will be considered correct if and only if the absolute error or relative error of your answer is less than 10^{-6}.

Example

standard input	standard output
2 3 3 2 4 3 2	13.765723680546197 12.734809553184123

Problem M. Trees in the Pocket

DreamGrid has just found two trees, both consisting of n vertices, in his right pocket. Each edge in each tree has its own weight. The i-th edge in the first tree has a weight of a_i, while the i-th edge in the second tree has a pair of weight, denoted by (b_i, c_i).

Let ${}^1u$ be the u-th vertex in the first tree, and ${}^2u$ be the u-th vertex in the second tree. Let $\mathbb{E}_1({}^1u, {}^1v)$ be the set containing the **indices** of all the edges on the path between the u-th and the v-th vertex in the first tree. Similarly, let $\mathbb{E}_2({}^2u, {}^2v)$ be the set containing the **indices** of all the edges on the path between the u-th and the v-th vertex in the second tree. Define the following values:

- $A_{\min}({}^1u, {}^1v) = \min\{a_k | k \in \mathbb{E}_1({}^1u, {}^1v)\}$
- $B_{\max}({}^2u, {}^2v) = \max\{b_k | k \in \mathbb{E}_2({}^2u, {}^2v)\}$
- $C_{\max}({}^2u, {}^2v) = \max\{c_k | k \in \mathbb{E}_2({}^2u, {}^2v)\}$

As DreamGrid is bored, he decides to count the number of good indices. DreamGrid considers an index i $(1 \le i \le n)$ is good, if for all $1 \le j \le n$ and $j \ne i$, $A_{\min}({}^1i, {}^1j) \ge \min(B_{\max}({}^2i, {}^2j), C_{\max}({}^2i, {}^2j))$. Please help DreamGrid figure out all the valid indices.

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 an integer n $(2 \le n \le 10^5)$, indicating the number of vertices in both trees.

For the following $(n-1)$ lines, the i-th line contains three integers ${}^1u_i$, ${}^1v_i$ and a_i $(1 \le {}^1u_i, {}^1v_i \le n$, $1 \le a_i \le 10^9)$, indicating that there is an edge, whose weight is a_i, connecting vertex u_i and v_i in the first tree.

For the following $(n-1)$ lines,the i-th line contains four integers ${}^2u_i$, ${}^2v_i$, b_i and c_i $(1 \le {}^2u_i, {}^2v_i \le n$, $1 \le b_i^2, c_i^2 \le 10^9)$, indicating that there is an edge, whose weight is (b_i, c_i), connecting vertex u_i and v_i in the second tree.

It's guaranteed that the sum of n of all test cases does not exceed 1.5×10^5.

Output

For each test case output k integers (k is the number of valid indices) in one line separated by a space in increasing order, indicating the indices of the valid vertices.

Note that if there is no valid vertex, you should print "-1" instead.

Example

standard input	standard output
2 2 1 2 1 1 2 2 3 4 1 2 7 1 3 8 1 4 12 1 2 8 8 2 3 9 7 2 4 6 4	-1 3 4