

Problem A

Best Contest Director

Time limit: 3 seconds

You might know that Little Cyan Fish creates a lot of contests every year. You know, Universal Cup contests, ICPC contests, Informatics Olympiads, and so on and so on... I have heard that Little Cyan Fish has been really tired these days, so I would like to help him.

Specifically, I have created n problem-setting groups, labeled from 1 to n . The i -th group initially has k_i members, whose strengths are the integers $a_{i,1}, a_{i,2}, \dots, a_{i,k_i}$. The **influence** of a group is the **sum** of the strengths of its members. An empty group has an influence of 0.

But Little Cyan Fish does not trust the members I have selected for him (which is sad). As the director, he wants to kick out as many of them as possible. However, he wants to be fair: whenever he kicks out a member, that member must have the smallest strength in their group. Formally, in one operation, Little Cyan Fish chooses a nonempty group and dismisses one of its members with the smallest strength. If several members tie, he may choose any of them.

You know, the members will be very sad to hear that someone has been fired, or... maybe they will even rise up in arms! Little Cyan Fish really does not want any problem-setting group to become dominant in the process. Therefore, **after each operation**, the influence of every group must be **no more than** the sum of the influences of all the other groups. That is, for every group i ,

$$s_i \leq \sum_{j \neq i} s_j,$$

where s_i denotes the current influence of group i . The initial groups do not necessarily satisfy this condition.

Find the maximum number of members that can be dismissed, and construct an optimal sequence of operations.

Input

There are multiple test cases. The first line contains a single integer T ($1 \leq T$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 5 \times 10^5$), the number of groups. Each of the next n lines describes one group: the i -th of these lines begins with an integer k_i ($1 \leq k_i \leq 5 \times 10^5$), followed by k_i integers $a_{i,1}, a_{i,2}, \dots, a_{i,k_i}$ ($1 \leq a_{i,j} \leq 10^9$), the strengths of the members of the i -th group.

It is guaranteed that the sum of n over all test cases does not exceed 5×10^5 , and the sum of k_i over all test cases does not exceed 5×10^5 .

Output

For each test case, output two lines. The first line contains a single integer m , the maximum number of members that can be dismissed. The second line contains m integers $b_1, b_2, \dots, b_m$ ($1 \leq b_i \leq n$), where b_i is the label of the group chosen in the i -th operation. If $m = 0$, the second line should be empty.

If there are multiple optimal sequences of operations, you may output any of them.

Sample Input 1

```
3
4
1 1
2 2 4
2 3 4
1 8
1
1 7
2
2 1 1
2 1 1
```

Sample Output 1

```
4
3 2 4 1
1
1
0
```

Problem B

Call You With Your Name 4

Time limit: 3 seconds

Little Cyan Fish likes string theory very much. Today, Little Cyan Fish invites you to continue studying human nicknames with him.

In Little Cyan Fish's universe, a human nickname can be represented as a string consisting only of lowercase Latin letters (a to z). For example, "qingyu" and "xiuga" are human nicknames, but "Abacde" is not.

Little Cyan Fish considers a name s to be a **Lyndon string** if and only if s is lexicographically smaller* than every proper suffix† t of s . For example, "abacde" is a Lyndon string, but "qingyu" is not a Lyndon string (because it is not lexicographically smaller than its proper suffix "ingyu").

Now, Little Cyan Fish gives you a human nickname s of length n . In one operation, you can choose one position and replace its character with any lowercase Latin letter. Find the minimum number of operations needed to make s a Lyndon string.

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^4$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 3000$). The second line contains a string s of length n , consisting only of lowercase Latin letters.

It is guaranteed that the sum of n^2 over all test cases does not exceed 9×10^6 .

Output

For each test case, output a single line containing a single integer, indicating the minimum number of operations needed to make s a Lyndon string.

Sample Input 1

Sample Output 1

4	0
4	1
abac	2
4	3
aaaa	
4	
bbaa	
5	
bbaaa	

In the first sample test case, the string is already a Lyndon string, so no operation is needed.

In the second sample test case, Little Cyan Fish can change the last character to b, obtaining aaab.

In the third sample test case, he can change the first and last characters to a and c, respectively, obtaining abac. One operation is not enough.

*A string is lexicographically smaller than another string if it is a proper prefix of the other string, or if its character at the first position where the strings differ comes earlier in the alphabet.

†A proper suffix is a non-empty suffix that is not equal to the original string. For example, "qqoj" has 3 proper suffixes, which are "j", "oj", and "qoj".

In the fourth sample test case, he can change the first two characters to a and the last character to b, obtaining aaaab. It is impossible to achieve this in fewer than three operations.

Problem C

Strange Sorting 3

Time limit: 4 seconds

Little Cyan Fish has a permutation $p_1, p_2, \dots, p_n$ of the integers from 1 to n , and, as the title of the problem suggests, he wants to sort it in ascending order. This time, the operation he can perform is as follows.

- In one operation, Little Cyan Fish may choose two integers i and j such that $1 \leq i < j \leq n$ and swap p_i and p_j .

But, come on, everyone knows how to sort a permutation with simple swaps, so obviously this is not what the task asks you to do. Instead, Little Cyan Fish defines the *distance* of an operation to be $j - i$. For a sequence of operations that sorts the permutation, let a_d denote the number of operations of distance d , for every $1 \leq d < n$.

Among all finite sequences of operations that sort the permutation, Little Cyan Fish wants you to find the lexicographically smallest sequence $(a_1, a_2, \dots, a_{n-1})$, and construct a sequence of **at most** $5n$ operations that achieves it.

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^6$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($2 \leq n \leq 10^6$), indicating the length of the permutation. The second line contains n integers $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq n$), indicating the permutation. It is guaranteed that p is a permutation of length n .

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

Output

For each test case, output the answer in the following format.

The first line for each test case should contain $n - 1$ integers $a_1, a_2, \dots, a_{n-1}$, the lexicographically smallest sequence.

The second line should contain a single integer k ($0 \leq k \leq 5n$), the number of operations in your sequence.

Each of the following k lines should contain two integers i and j ($1 \leq i < j \leq n$), the indices chosen in the corresponding operation. The operations, performed in the given order, must sort the input permutation, and for every $1 \leq d < n$, exactly a_d of the operations must have distance d .

It can be shown that an optimal sequence of at most $5n$ operations always exists. If there are multiple valid sequences of operations, you may output any of them.

Sample Input 1**Sample Output 1**

```
3
3
3 2 1
4
2 1 3 4
4
1 3 2 4
```

```
0 1
1
1 3
0 1 2
3
1 4
2 4
1 4
0 3 2
5
1 3
1 4
2 4
1 4
1 3
```

In the first sample test case, Little Cyan Fish swaps positions 1 and 3. This sorts the permutation using one operation of distance 2.

In the second sample test case, one optimal sequence is (1, 4), (2, 4), (1, 4). The distances are 3, 2, and 3, so the resulting sequence is (0, 1, 2).

In the third sample test case, one optimal sequence is (2, 4), (1, 4), (1, 3), (1, 4), (2, 4). Its distance counts are (0, 3, 2).

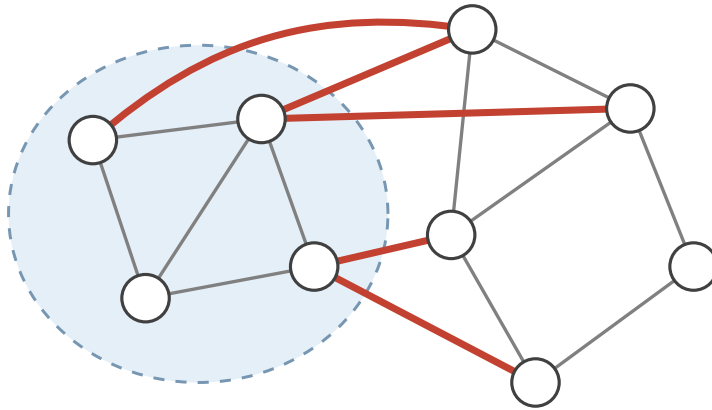
Problem D

Guess the Graph

Time limit: 1 seconds

This is an interactive problem.

Little Cyan Fish has a hidden simple undirected graph with n vertices, labeled from 1 to n . You know the value of n , but you do not know any of the edges. Little Cyan Fish loves the concept of a *cut*. For a set of vertices S , let $\text{cut}(S)$ denote the number of edges with exactly one endpoint in S .



Little Cyan Fish allows you to compare cuts by making queries. In one query, you choose two sets of vertices S_1 and S_2 . Little Cyan Fish tells you whether $\text{cut}(S_1)$ is smaller than, equal to, or greater than $\text{cut}(S_2)$.

You can probably predict that the next token will be something about recovering the graph, but this time the task is easier. Little Cyan Fish gives you a vertex u , and he wants you to find all vertices adjacent to u in at most $n + 10$ queries.

Input

The first line of the input contains two integers n and u ($1 \leq n \leq 1000$, $1 \leq u \leq n$), indicating the number of vertices and the given vertex, respectively. Then, the interaction begins.

Interaction Protocol

To make a query, print a line in the following format:

? s_1 s_2

where s_1 and s_2 are strings of length n , consisting only of characters 0 and 1. For each $j \in \{1, 2\}$, vertex i belongs to S_j if and only if the i -th character of s_j is 1. The sets may be empty, may overlap, and may be equal.

As a response to the query, you will get a single character $<$, $=$, or $>$, meaning that $\text{cut}(S_1)$ is less than, equal to, or greater than $\text{cut}(S_2)$, respectively. You must ensure that the number of queries you make does not exceed $n + 10$.

To report the answer, output a line in the following format:

! k v_1 v_2 $\dots$ v_k

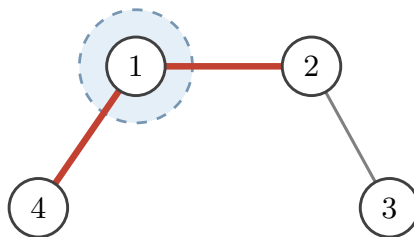
where k ($0 \leq k \leq n - 1$) is the number of vertices adjacent to u , and $v_1, v_2, \dots, v_k$ are their distinct labels in any order. If $k = 0$, output ! 0. Reporting the answer does not count as a query. Your program must terminate immediately after reporting the answer.

After outputting a query or reporting the answer, do not forget to output a newline character and flush the output stream. If you make an invalid query or exceed the query limit, the interactor will close the input stream and your program must terminate.

In this problem, the interactor is adaptive. This means that the graph may change depending on your queries, as long as it does not contradict any previous response. Your answer must be correct for every graph consistent with all responses you have received.

Read	Sample Interaction 1	Write
4 1		
	? 0000 1000	
<		
	? 0100 1100	
=		
	? 0010 1010	
<		
	? 0001 1001	
=		
	! 2 2 4	

One graph consistent with the sample interaction has edges (1, 2), (1, 4), and (2, 3).



Testing Tool: A testing tool is provided to help contestants develop and test their solutions. You can download this tool from the attachments. Run the tool with the `-h` option to display usage instructions. The tool checks the interaction and the reported answer, but does not enforce the time or memory limit.

The testing tool uses a fixed graph specified in its input file throughout the interaction, whereas the official interactor is adaptive. Passing tests with the testing tool does not guarantee that your solution is correct against the official interactor.

Problem E

Subset Sums

Time limit: 1 seconds

After solving the Subset Sum problem in $O(2^{0.49n})$ time, Little Cyan Fish wants you to solve the following task.

Little Cyan Fish has n cards. The i -th card ($1 \leq i \leq n$) has a non-negative integer a_i written on it.

For each subset $S \subseteq [n]$ of card indices, where $[n] = \{1, 2, \dots, n\}$, he records the sum of the integers on the chosen cards, i.e., $\sum_{i \in S} a_i$. The sum for the empty subset is 0. Different subsets are recorded separately, even if their sums are equal.

After considering all 2^n subsets, he sorts these 2^n sums in non-decreasing order and denotes them by $s_0, s_1, \dots, s_{2^n-1}$. Little Cyan Fish wants you to find the maximum difference between two consecutive sums, that is,

$$\max_{0 \leq i < 2^n - 1} (s_{i+1} - s_i).$$

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 10^6$), indicating the number of cards. The second line contains n integers $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq 10^9$), indicating the integers written on the cards.

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

Output

For each test case, output a single line containing a single integer, indicating the maximum difference between two consecutive sums.

Sample Input 1

Sample Output 1

4	0
1	5
0	1
3	4
2 3 10	
3	
1 1 3	
3	
0 4 4	

In the first sample test case, both subset sums are 0, so the answer is 0.

In the second sample test case, the sorted subset sums are 0, 2, 3, 5, 10, 12, 13, 15. The maximum difference is 5, between 5 and 10.

In the third sample test case, the sorted subset sums are 0, 1, 1, 2, 3, 4, 4, 5. The answer is 1.

In the fourth sample test case, the sorted subset sums are 0, 0, 4, 4, 4, 4, 8, 8. The answer is 4.

Problem F

Mode

Time limit: 3 seconds

Little Cyan Fish has an $n \times n$ matrix W .

For an integer sequence $a_1, a_2, \dots, a_n$ with $1 \leq a_i \leq n$, let b_i denote the number of distinct integers that occur the most times in the prefix $a_1, a_2, \dots, a_i$. For example, in the prefix 1, 2, 1, 2, 3, both 1 and 2 occur twice, so $b_5 = 2$.

Little Cyan Fish defines the weight of the sequence to be

$$\prod_{i=1}^n w_{i, b_i}.$$

Find the sum of the weights of all such sequences. Since the answer can be huge, output it modulo 998244353.

Input

The first line of the input contains a single integer n ($1 \leq n \leq 100$), indicating the length of the sequences and the size of the matrix.

The next n lines describe the matrix W . The i -th of these lines contains n integers $w_{i,1}, w_{i,2}, \dots, w_{i,n}$ ($0 \leq w_{i,j} < 998244353$).

Output

Output a single line containing a single integer, indicating the sum of the weights of all such sequences modulo 998244353.

Sample Input 1

1 7	7
--------	---

Sample Output 1

Sample Input 2

2 1 2 3 4	14
-----------------	----

Sample Output 2

Sample Input 3

3 1 1 1 3 1 1 0 0 1	6
------------------------------	---

Sample Output 3

In the first sample test case, the only sequence is (1), and its weight is 7.

In the second sample test case, the sequences (1, 1) and (2, 2) each have weight 3, while (1, 2) and (2, 1) each have weight 4. The sum is $3 + 3 + 4 + 4 = 14$.

In the third sample test case, a sequence has nonzero weight only if all three values are different. There are six such sequences, each with weight 1.

Problem G

Teleport

Time limit: 4 seconds

Little Cyan Fish is on a number line. Initially, he is at position a_0 , and his home is at position 0.

He wants to visit the positions $a_1, a_2, \dots, a_n$ in this order. He can walk in either direction along the number line. At any time, he can teleport to his home or move his home to his current position. Both actions are free and can be performed any number of times, including while he is walking.

Find the minimum total distance he needs to walk to visit all the positions in the given order.

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 5 \times 10^5$), indicating the number of positions to visit. The second line contains $n + 1$ integers $a_0, a_1, \dots, a_n$ ($-10^9 \leq a_i \leq 10^9$), indicating the initial position and the positions to visit.

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

Output

For each test case, output a single line containing a single integer, indicating the minimum total distance Little Cyan Fish needs to walk.

Sample Input 1

Sample Output 1

4	0
1	10
7 0	6
2	8
0 10 5	
3	
0 4 2 0	
3	
-2 3 -2 3	

In the first sample test case, Little Cyan Fish teleports to his home at position 0, so he does not need to walk.

In the second sample test case, he walks from 0 to 10 and moves his home to position 5 on the way. He then teleports to position 5. The total distance is 10.

In the third sample test case, he walks from 0 to 4 and moves his home to position 2 on the way. He then teleports to 2 and walks to 0. The total distance is $4 + 2 = 6$.

In the fourth sample test case, he teleports to 0, walks to 3, and moves his home to 3. He then walks to -2 and teleports to 3. The total distance is $3 + 5 = 8$.

Problem H

Idols and Fans

Time limit: 4 seconds

Guess what? In the year 6202, Little Cyan Fish's server is welcoming its 10^6 -th member! Never heard of it? Well, the server, also known as Qingyu's group, is where his friends hang out and talk about CP, non-CP, PKU jokes, food in Beijing, and just about everything else. But most importantly, the chat is full of messages every day about who is whose idol.

Following tradition, Little Cyan Fish is organizing an idol party with n members, numbered from 1 to n . There are m fan-idol relationships among these members: in the i -th relationship, member u_i is a fan of member v_i , and a gift from u_i to v_i costs w_i . At the party, each member must give **exactly one** special gift to **one of their idols**. However, two members exchanging gifts with each other would be awkward (although that sounds pretty sweet :p), so no pair of members may give gifts to each other.

Find the minimum total cost of the gifts, or determine that no valid arrangement exists.

Input

The first line contains an integer T ($1 \leq T$), the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers n and m ($1 \leq n \leq 10^6, 0 \leq m \leq 10^6$), the number of members and the number of fan-idol relationships, respectively.

The i -th of the next m lines contains three integers u_i, v_i , and w_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i, 0 \leq w_i \leq 10^9$), where member u_i is a fan of member v_i , and a gift from u_i to v_i costs w_i .

It is guaranteed that the sum of n over all test cases does not exceed 10^6 , and the sum of m over all test cases does not exceed 10^6 . It is guaranteed that each ordered pair (u_i, v_i) appears at most once in a test case. Two members may be fans of each other.

Output

For each test case, output an integer — the minimum possible total cost of the gifts. If there is no solution, output -1 . Output each answer on a separate line.

Sample Input 1

Sample Output 1

3	6
3 3	-1
1 2 1	1
2 3 2	
3 1 3	
2 2	
1 2 0	
2 1 0	
3 4	
1 2 0	
1 3 1	
2 1 0	
3 2 0	

In the first sample test case, member 1 gives a gift to member 2, member 2 to member 3, and member 3 to member 1. The total cost is $1 + 2 + 3 = 6$.

In the second sample test case, members 1 and 2 would have to give gifts to each other, so no valid arrangement exists.

In the third sample test case, member 1 gives a gift to member 3, member 2 to member 1, and member 3 to member 2. The total cost is 1.

Problem I

Graph Game

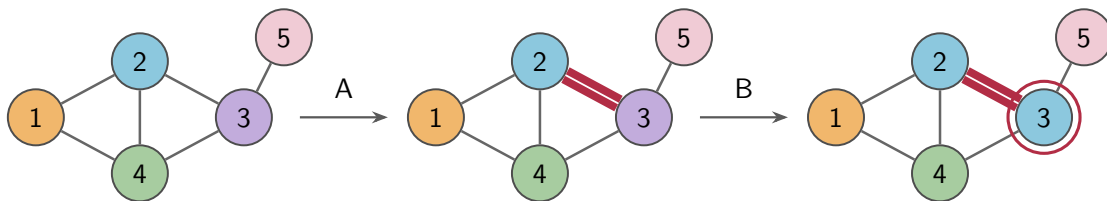
Time limit: 3 seconds

This is an interactive problem.

Little Cyan Fish has a simple undirected graph with n vertices labeled from 1 to n and m edges labeled from 1 to m . Initially, the color of vertex i is i .

Two players, A and B, play a game consisting of m rounds. In each round, the players take the following actions in order:

- Player A chooses an edge (u, v) that has not been chosen in any previous round.
- Player B chooses one endpoint of this edge and changes its color to the current color of the other endpoint.



After all m rounds, Player B wins if all vertices have the same color. Otherwise, Player A wins.

Little Cyan Fish wants you to determine which player can guarantee a win regardless of the other player's moves. Then play as that player against him and win the game.

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^4$), indicating the number of test cases.

For each test case, the first line contains two integers n and m ($2 \leq n \leq 10^5$, $1 \leq m \leq 10^5$), indicating the number of vertices and the number of edges, respectively.

The next m lines describe the edges. The i -th of these lines contains two integers u_i and v_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$), meaning that edge i connects vertices u_i and v_i .

It is guaranteed that the graph has no self-loops or multiple edges. The graph may be disconnected.

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

After reading a graph, complete the interaction for that test case before reading the next graph.

Interaction Protocol

To choose your role, output a line containing A or B. Output A if Player A can guarantee a win, or B if Player B can guarantee a win. Little Cyan Fish will play as the other player.

Then play exactly m rounds as follows.

- If you play as A, output a line containing a single integer e ($1 \leq e \leq m$), the index of an edge that has not been chosen in any previous round. As a response, you will get a single integer x on a line. Vertex x is an endpoint of edge e , and its color changes to the current color of the other endpoint.

- If you play as B, read a single integer e ($1 \leq e \leq m$) on a line, the index of an edge that has not been chosen in any previous round. Output a line containing a single integer x , an endpoint of edge e . The color of vertex x changes to the current color of the other endpoint.

After all m rounds, read a single integer. This integer is 1 if you chose the correct role and won the game. After receiving 1, continue with the next test case. After the last test case, your program should terminate.

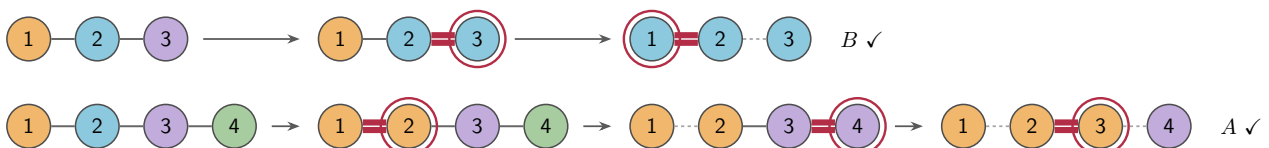
The interactor will send -1 if you choose an incorrect role, make an invalid move, or lose a game. If you receive -1 at any point, your program should terminate immediately.

After outputting your role or a move, do not forget to output a newline character and flush the output stream.

Testing Tool: A testing tool is provided to help contestants develop and test their solutions. You can download this tool from the attachments. Run the tool with the $-h$ option to display usage instructions. The testing tool covers only some of the test scenarios and features supported by the official interactor.

Read	Sample Interaction 1	Write
2		
3 2		
2 3		
1 2		
	B	
1		
	3	
2		
	1	
1		
4 3		
1 2		
2 3		
3 4		
	A	
	1	
2		
	3	
4		
	2	
3		
1		

The two games in the sample are shown below. In each step, A selects the highlighted edge and B recolors the circled vertex. Dashed edges were chosen earlier.



Problem J

Merge Sequence

Time limit: 2 seconds

Little Cyan Fish has m sequences consisting of n integers in total. Each integer from 1 to n occurs exactly once across all the sequences.

He wants to merge these sequences into a permutation p of length n . Initially, p is empty. In each step, he chooses a nonempty sequence, removes its first integer, and appends this integer to the end of p . He repeats this until all the sequences are empty.

An integer is called a fixed point of p if its value equals its position in p . Positions are numbered from 1 to n .

Find the minimum possible number of fixed points in p .

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^6$), indicating the number of test cases.

For each test case, the first line contains two integers n and m ($1 \leq m \leq n \leq 10^6$), indicating the total number of integers and the number of sequences, respectively.

The next m lines describe the sequences. The i -th of these lines contains an integer k_i ($1 \leq k_i \leq n$), followed by k_i integers $a_{i,1}, a_{i,2}, \dots, a_{i,k_i}$ ($1 \leq a_{i,j} \leq n$), describing the i -th sequence in order.

It is guaranteed that the sum of k_i over all sequences in each test case is n , and that each integer from 1 to n occurs exactly once across all the sequences.

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

Output

For each test case, output a single line containing a single integer, indicating the minimum possible number of fixed points in p .

Sample Input 1

4	1
1 1	0
1 1	1
3 2	0
1 1	
2 2 3	
3 2	
2 1 3	
1 2	
4 2	
3 2 1 3	
1 4	

Sample Output 1

In the first sample test case, the only possible permutation is $[1]$, which has one fixed point.

In the second sample test case, Little Cyan Fish can obtain $p = [2, 3, 1]$, which has no fixed points.

In the third sample test case, $p = [1, 3, 2]$ is an optimal permutation with one fixed point.

In the fourth sample test case, he can obtain $p = [2, 4, 1, 3]$, which has no fixed points.

Problem K

Iridescent Universe 3

Time limit: 4 seconds

The Universe of Cup is where Little Cyan Fish lives. On the biggest planet in the universe, called Rainbow-Earth, everything is painted in one of k colors, numbered from 1 to k . Little Cyan Fish has n items, and the i -th item has color a_i and value b_i .

As the Chief Executive, Little Cyan Fish is preparing welcome gifts for the contestants of The 4th Universal Cup Finals. He loves a colorful world, so he wants to choose exactly k items, one of each color.

For some reason (perhaps because he likes the term, or simply because he does not know what else to ask you), Little Cyan Fish defines the *score* of a selection as the sum of the absolute differences between the values of every unordered pair of chosen items. Formally, for a set of chosen indices $S \subseteq \{1, 2, \dots, n\}$ with $|S| = k$ and $\{a_i \mid i \in S\} = \{1, 2, \dots, k\}$, its score is

$$\sum_{\substack{i, j \in S \\ i < j}} |b_i - b_j|.$$

Find the maximum possible score.

Input

The first line contains two integers n and k ($1 \leq n \leq 2 \times 10^5$, $1 \leq k \leq \min(n, 2000)$), indicating the number of items and the number of colors, respectively.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq k$), where a_i is the color of the i -th item.

The third line contains n integers $b_1, b_2, \dots, b_n$ ($0 \leq b_i \leq 10^9$), where b_i is the value of the i -th item.

It is guaranteed that every integer from 1 to k appears in array a at least once.

Output

Output a single line containing a single integer, indicating the maximum possible score.

Sample Input 1

```
4 2
1 1 2 2
0 2 1 3
```

Sample Output 1

```
3
```

Sample Input 2

```
6 3
1 2 1 3 2 3
1 7 9 4 2 6
```

Sample Output 2

```
14
```

Sample Input 3

```
8 4
1 1 2 2 3 3 4 4
0 1 0 1 0 1 0 1
```

Sample Output 3

```
4
```

In the first sample test case, Little Cyan Fish can choose indices 1 and 4, giving a score of $|0 - 3| = 3$.

In the second sample test case, choosing indices 3, 5, and 4 gives values 9, 2, and 4, with a score of $|9 - 2| + |9 - 4| + |2 - 4| = 14$.

In the third sample test case, Little Cyan Fish can choose values 0, 0, 1, and 1, one from each color. The four pairs with different values each contribute 1, so the maximum score is 4.

Problem L

White Night 3

Time limit: 3 seconds

Little Cyan Fish wants to color n points cyan or white. The i -th point is at (x_i, y_i) . The points are pairwise distinct and are the vertices of a strictly convex polygon, listed in counterclockwise order.

Let H_c be the convex hull of the cyan points, and let H_w be the convex hull of the white points. A convex hull is the smallest convex set containing the given points. The convex hull of an empty set is empty; a convex hull consisting of a point or a line segment has area zero.

Find the maximum possible area of $H_c \cap H_w$, the region common to the two convex hulls.

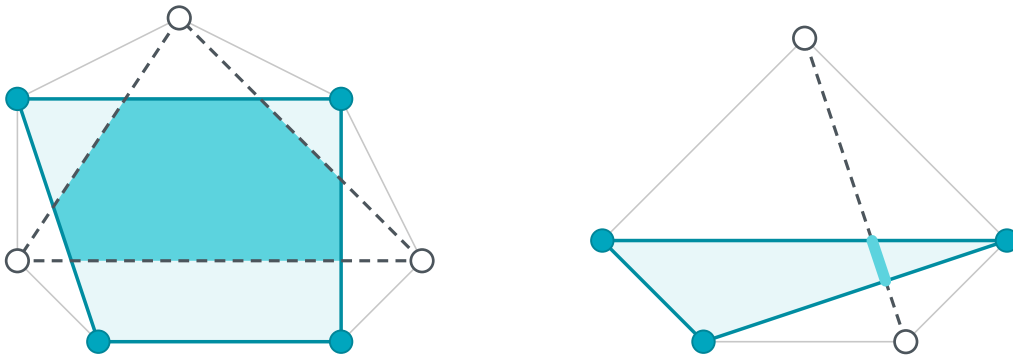


Illustration of the first two sample test cases (first on the left, second on the right).

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^4$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($3 \leq n \leq 10^5$), indicating the number of points. Each of the following n lines contains two integers x_i and y_i ($-10^9 \leq x_i, y_i \leq 10^9$), indicating the coordinates of the i -th point.

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

Output

For each test case, output a single line containing a real number, indicating the maximum possible area of the intersection of the two convex hulls.

Your answer will be accepted if its absolute or relative error does not exceed 10^{-8} . Formally, let your output be a and the jury's answer be b ; your output is accepted if and only if $\frac{|a-b|}{\max(1, |b|)} \leq 10^{-8}$.

Sample Input 1**Sample Output 1**

```
3
7
0 0
3 0
4 1
3 3
1 4
-1 3
-1 1
5
0 0
2 0
3 1
1 3
-1 1
7
-1000000000 -1000000000
-5000000001 -5000000002
4999999998 4999999995
9999999998 9999999994
5000000000 4999999997
-4999999999 -5000000000
-9999999999 -9999999999
```

```
5.944444444444444444444444444444
0
2.458333333333333333333333333333404
```

In the first sample test case, Little Cyan Fish colors points 1, 2, 4, and 6 cyan, and the other points white. The cyan quadrilateral and the white triangle overlap in an area of $\frac{107}{18}$, which is the maximum possible area.

In the second sample test case, at least one color has at most two points, so the intersection has area zero. The figure shows points 1, 3, and 5 colored cyan; the white convex hull is the line segment joining points 2 and 4.

In the third sample test case, coloring points 1, 4, and 6 cyan and the other points white gives the maximum intersection area of $\frac{59}{24}$.

Problem M

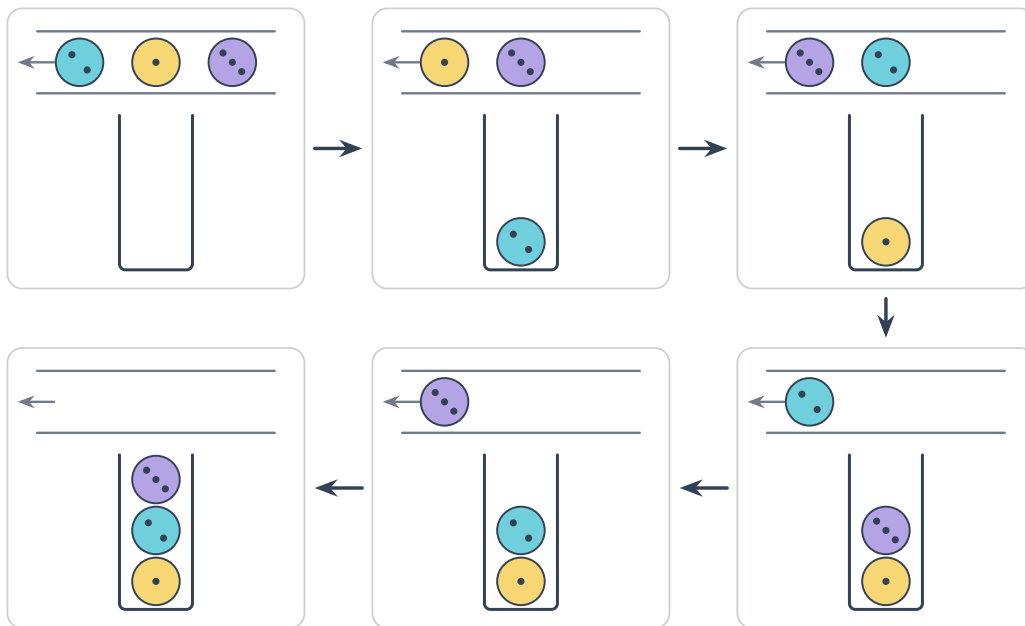
Queue Editor 2

Time limit: 2 seconds

To celebrate the fact that no team solved the problem “Queue Editor” at The 3rd Universal Cup Semifinals, Little Cyan Fish has come up with a new problem involving a queue and... this time, a stack! He has n balls labeled from 1 to n . The balls are initially arranged in a queue, and he also has an empty stack. While the queue is not empty, he performs the following steps:

1. Remove the ball at the front of the queue. Let x be its label.
2. While the stack is not empty and the label of its top ball is greater than x , remove the top ball from the stack and append it to the back of the queue. The balls are appended in the order in which they are removed.
3. Push the ball labeled x onto the top of the stack.

It can be shown that the queue becomes empty after finitely many steps. Find the total number of times Little Cyan Fish pushes a ball onto the stack before this happens.



Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($1 \leq n \leq 5 \times 10^5$), indicating the number of balls. The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$), indicating the labels of the balls in the initial queue from front to back. It is guaranteed that $a_1, a_2, \dots, a_n$ form a permutation of $1, 2, \dots, n$.

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

Output

For each test case, output a single line containing a single integer, indicating the total number of times a ball is pushed onto the stack.

Sample Input 1

4
1
1
3
1 2 3
3
2 1 3
3
3 1 2

Sample Output 1

1
3
5
4

Sample Input 2

2
3
2 3 1
6
1 3 4 6 2 5

Sample Output 2

6
15

In the first example, the sequences of labels pushed onto the stack are (1), (1, 2, 3), (2, 1, 3, 2, 3), and (3, 1, 2, 3) in the four test cases, respectively. The figure in the problem statement illustrates the third test case in the first example.

In the first test case of the second example, the sequence of pushed labels is (2, 3, 1, 3, 2, 3), so the answer is 6. In the second test case of the second example, the sequence is (1, 3, 4, 6, 2, 5, 6, 4, 3, 6, 5, 4, 6, 5, 6), so the answer is 15.