The 10th China Collegiate Programming Contest Finals

Contest Session

May 11, 2025

This problem set contains 13 problems and spans 22 pages.

ID	English Title	Chinese Title
A	**R**equiem for Qingyu	青鱼安魂曲
B	**A**dd One 3	加一 3
C	**I**ridescent Universe	彩虹色的宇宙
D	**N**octurne without a Moon	无月夜色
E	**O**mniscient Artist	全知艺术家
F	**W**itnessing the Miracle	见证奇迹
G	$+$ and $\times$ with a sugar	$+$, $\times$ 与糖
H	**Q**ingyu's Little Training Center	小青鱼的训练中心
I	**I**neffable Cycle	不可言喻的环
J	**N**ot a work of Idol	不是偶像的大作
K	**G**rotesque Team Reconstruction	怪诞组队法
L	**Y**earning for Yonder	对远方的向往
M	**U**nder the Epilogue	尾声之下

Problem A. Requiem for Qingyu

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1.5 seconds
Memory limit:	1024 megabytes

The background story in the Chinese statements is removed due to the translation difficulties. >_<

You are given a string S. Little Cyan Fish wants to divide S into several substrings $S = U^{(1)}U^{(2)} \dots U^{(k)}$. Here, each $U^{(i)}$ must be a non-empty string, and the concatenation of all $U^{(i)}$ gives back the original string S.

However, that's not enough. Little Cyan Fish wants to find a favorite string T from within the requiem. A string T is defined as Little Cyan Fish's favorite if and only if there exists a partition of $S = U^{(1)}U^{(2)} \dots U^{(k)}$ such that every $U^{(i)}$ $(1 \le i \le k)$ is a prefix of T.

Now, your task is: given a string S, compute how many substrings of S are Little Cyan Fish's favorite. In other words, count the number of pairs (l, r) such that $1 \le l \le r \le |S|$ and the substring $S[l, r]$ is Little Cyan Fish's favorite.

Recall that $S[l, r]$ refers to the substring of S starting from position l to r (inclusive), that is, $S_l S_{l+1} \dots S_r$.

Input

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

The first line of the input contains a string S $(1 \le |S| \le 10^5)$, which represents Little Cyan Fish's requiem. The requiem only contains lowercase Latin letters (from "`a`" to "`z`", a total of 26 characters).

It is guaranteed that the sum of $|S|$ across all test cases does not exceed 2×10^5.

Output

For each test case, output a single line contains a single integer, indicating the answer.

Example

standard input	standard output

`4`
`aaaa`
`abbaabba`
`abaababa`
`qingyuislittlecyanfish`
	`10`
	`8`
	`13`
	`1`

Problem B. Add One 3

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

抓住和抓不住的照片 哪张更美
去过和没去过的地方 哪里更远

白鸟过河滩 by ilem

The background story in the Chinese statements is removed due to the translation difficulty >_ <

In computer science, the maximum subarray sum problem, also known as the maximum contiguous subarray problem, is the task of finding a contiguous subarray within a one-dimensional array $A_1, A_2, \cdots, A_n$ that has the largest sum. Formally, the task is to find indices i and j such that the following sum is as large as possible:

$$\sum_{i \le k \le j} A_k$$

You can also choose an empty array, which means you have found a subarray with a sum of 0. The value of the maximum subarray sum is denoted as $MSS(A)$. For example, $MSS([-2, 1, 4, -3, 5]) = 7$, $MSS([-5]) = 0$, $MSS([-1, -2]) = 0$.

In the Little Cyan Fish's heart, there is an integer sequence of length n, $a_1, a_2, \ldots, a_n$. You can perform several operations, each time choosing an element a_i and incrementing its value, i.e., updating $a_i \leftarrow a_i + 1$.

The Little Cyan Fish has many questions in his mind. This time, he wants you to answer q queries, each time giving you a range $[l, r]$ and asking how many operations are needed at a minimum to make $a_l, a_{l+1}, \ldots, a_r$ the **unique** maximum subarray sum of the entire sequence. If it is not possible to achieve this with any number of operations, then output -1. Since we consider not selecting a subarray as a solution with a sum of 0, the final unique maximum subarray sum must be greater than 0.

Of course, the Little Cyan Fish will only think in his mind and will not actually take action. Therefore, each query is independent.

Input

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

- The first line contains two integers n, q ($1 \le n, q \le 5 \times 10^5$), representing the length of the sequence and the number of queries.
- The next line contains n integers $a_1, a_2, \ldots, a_n$ ($-10^9 \le a_i \le 10^9$).
- The next q lines each contain two integers l, r ($1 \le l \le r \le n$), representing the left and right endpoints of a query.

It is guaranteed that the sum of n across all test data does not exceed 5×10^5, and the sum of q does not exceed 5×10^5.

Output

For each set of test data, output q lines, each containing an integer representing the answer to each query.

Example

standard input	standard output
2 6 6 -1 2 -2 0 1 -1 1 1 1 2 1 3 1 4 1 5 1 6 1 1 0 1 1	-1 2 -1 -1 4 5 1

Problem C. Iridescent Universe

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 512 megabytes

雨过天晴白云飘，
天空架起彩虹桥。
赤橙黄绿青蓝紫，
数数颜色有七道。

The background story in the Chinese statements is removed due to the translation difficulties. >_<

When Little Cyan Fish was young, he believed that formal science is also a form of art. Just as an artist defines his canvas, theoretical scientists also use their own ways to give unique colors to theories.

Now, Little Cyan Fish has an undirected graph G with n nodes and m edges. In Little Cyan Fish's universe, there are k types of colors $1, 2, \cdots, k$. The i-th edge (where $1 \leq i \leq m$) connects nodes u_i and v_i, and its color is c_i (where $1 \leq c_i \leq k$). Of course, it is possible that there is a color that is not used by any edge—this is unavoidable, as some colors are destined not to appear in the world.

"How wonderful it is, from crimson to light blue, from grayish white to deep black." Unfortunately, Little Cyan Fish discovers that his favorite color does not appear in this universe. Little Cyan Fish can only settle for the k colors available to complete his work. After careful consideration, Little Cyan Fish believes that if the color of the i-th edge can ultimately be modified to t_i, then this world can still become very cool. To achieve this dream, Little Cyan Fish can use the following magic to modify this rainbow graph:

- Choose a node x and a color y, and color all edges adjacent to x (i.e., having one endpoint as x) with y.

Of course, Little Cyan Fish's magical power is limited; he can cast magic at most n times. Little Cyan Fish wants to know if it is possible to use magic to make the color of the i-th edge turn into color t_i. If possible, please provide Little Cyan Fish with a solution.

To help you solve this problem, Little Cyan Fish has proven that if there exists a valid solution, then there is certainly a solution that requires at most n operations. I hope this observation can help you and help Little Cyan Fish.

Input

The first line of input contains three integers n, m, k ($1 \leq n \leq 2 \times 10^5$, $0 \leq m \leq 2 \times 10^5$, $1 \leq k \leq 2 \times 10^5$), representing the number of nodes, the number of edges, and the number of color types, respectively.

The next m lines, the i-th line contains four integers u_i, v_i, c_i, t_i ($1 \leq u_i, v_i \leq n$, $1 \leq c_i, t_i \leq k$) representing the information of the i-th edge.

It is guaranteed that there are no multiple edges or self-loops in the given graph.

Output

If there is a solution, the first line of output should contain an integer s indicating the number of magic casts, ensuring that $0 \leq s \leq n$. The next s lines should each contain two integers u, c (ensuring that $1 \leq u \leq n$, $1 \leq c \leq k$) representing one operation.

If there is no solution, output a line with -1.

Examples

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

Problem D. Nocturne without a Moon

Input file: `standard input`
Output file: `standard output`
Time limit: 3.5 seconds
Memory limit: 1024 megabytes

> I am not wrong; I know who I am talking to. The child in the cradle will be greater in the future, and he will pass down the throne for several generations.

The background story in the Chinese statements is removed due to the translation difficulties. >_<

Little Cyan Fish gives you a permutation of size n, denoted by $p_1, p_2, \ldots, p_n$.

Little Cyan Fish wants to find the number of indices $1 \le i < j < k < l \le n$ such that $p_i \cdot p_k = p_j \cdot p_l$.

For example, for the permutation $p = [1\ 5\ 3\ 6\ 2\ 4]$, if we select the positions [**1** 5 **3** **6** **2** 4], then we have $1 \cdot 6 = 3 \cdot 2$.

At that time, the Little Cyan Fish wanted to know how many different ways he could choose such a quadruple (i, j, k, l) that would satisfy his requirement.

Input

The first line of input contains an integer n ($1 \le n \le 50\,000$).

The next line contains n integers $p_1, p_2, \ldots, p_n$ ($1 \le p_i \le n$). It is guaranteed that p is a permutation of numbers from 1 to n.

Output

Output a single integer, indicating the answer.

Example

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

Note

In the sample test case, there are four ways:

- [**5** 8 **1** **2** 4 3 6 **10** 9 7], because $5 \cdot 2 = 1 \cdot 10$
- [**5** 8 1 **2** **4** 3 6 **10** 9 7], because $5 \cdot 4 = 2 \cdot 10$
- [**5** 8 1 2 4 **3** **6** **10** 9 7], because $5 \cdot 6 = 3 \cdot 10$
- [5 **8** 1 2 **4** **3** **6** 10 9 7], because $8 \cdot 3 = 4 \cdot 6$

Problem E. Omniscient Artist

Input file: `standard input`
Output file: `standard output`
Time limit: 8 seconds
Memory limit: 512 megabytes

The background story in the Chinese statements is removed due to the translation difficulties. >_<

In addition to “adding one,” there were many other experiences during the premiere of Little Cyan Fish. Various data structure problems appeared at the premiere. From "return" to "WBLT," these problems were simply filled with dark humor —— or rather, their appearance merely reduced the number of problems in this scene by 1.

At that time, Little Cyan Fish’s collaborator wanted to become an omniscient artist of data structures. On many nights, he found Little Cyan Fish by the stove to talk, sharing many stories and making many wishes. Regardless of what happened at that time, Little Cyan Fish only remembers arguing with him about whether data structure problems should appear in this competition.

In the blink of an eye, three years have passed, and Little Cyan Fish has gradually forgotten the content of both sides’ arguments. Whether this collaborator is omniscient, Little Cyan Fish cannot know, but he indeed believes that he is not an artist. Perhaps different things have different kinds of romance, and not all humans can appreciate it.

That being said, this time, Little Cyan Fish wants you to experience the beauty of data structures. Now, Little Cyan Fish has given you n axis-parallel rectangular regions, with the i-th rectangle represented by four integers $x_{1,i}, x_{2,i}, y_{1,i}, y_{2,i}$.

In Little Cyan Fish’s mind, there is a fixed positive integer m ($m \geq \sqrt{n}$). For each integer c such that $1 \leq m \cdot c \leq n$, Little Cyan Fish needs you to calculate the area of the region that is exactly contained by $m \cdot c$ rectangles.

Formally, the area of the region that is exactly contained by i rectangles is the number of integer points (x, y) that satisfy $\sum_{j=1}^{n} [x_{1,j} \leq x < x_{2,j}][y_{1,j} \leq y < y_{2,j}] = i$.

Input

The first line of input contains two integers n, m ($2 \leq n \leq 3 \times 10^5, \sqrt{n} \leq m \leq n$).

The next n lines each contain four integers, $x_{1,i}, x_{2,i}, y_{1,i}, y_{2,i}$ ($1 \leq x_{1,i} < x_{2,i} \leq n$, $1 \leq y_{1,i} < y_{2,i} \leq n$).

Output

Output a total of $\lfloor \frac{n}{m} \rfloor$ lines, each representing the area of the region that is exactly contained by $m, 2m, 3m, \ldots, \lfloor \frac{n}{m} \rfloor \cdot m$ rectangles.

Examples

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

Problem F. Witnessing the Miracle

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

The background story in the Chinese statements is removed due to the translation difficulties. >_<

In Riyadh, Little Cyan Fish became an evangelist for a local programming competition. This was the first time Little Cyan Fish experienced the process of children from a different cultural circle exploring knowledge. When one enters a competition as a judge, their perspective on observing the event also changes. Little Cyan Fish began to recall their own first competition experience, thinking back to how they weren't even proficient at writing a simple segment tree and would feel immense joy from solving the easiest problems. Little Cyan Fish was always grateful for their past ignorance, which prevented anxiety about the future and allowed for that period of pure competitive experience.

After the competition, at the Irqah venue, the organizers provided the contestants with a science game experience. In this science game, the children received several magnets and arranged them on a number line. The coordinate of each magnet was a positive integer between $[1, n]$, and no two magnets occupied the same coordinate.

In one operation, a child could choose a magnet, activate it, and then remove it. Due to the repulsive forces between magnets, after activating a certain magnet, magnets to its left would move 1 unit to the left, and magnets to its right would move 1 unit to the right.

Little Cyan Fish had, after all, lost their childlike innocence and couldn't appreciate the fun the children had playing with the magnets. However, Little Cyan Fish noticed that these children, while randomly playing with these magnets, were very insistent on ensuring that all magnets eventually ended up back within the range $[1, n]$. Specifically, given a non-negative integer k, a legal experiment is defined as: performing exactly k operations, and satisfying the condition that in the final state, all magnet coordinates are still positive integers between $[1, n]$. **Little Cyan Fish wants you to note: each operation must select exactly one magnet, and during the process, magnet coordinates can temporarily be outside the range of positive integers $[1, n]$.**

Little Cyan Fish can use a binary string of length n (consisting of `0`s and `1`s) to represent the initial and final states of the experiment. Specifically, the i-th character of the string is `1` if and only if there is a magnet at coordinate i.

Little Cyan Fish watched with great interest as the children played with the magnets and suddenly thought of this problem: Given two strings S and T composed of `0`, `1`, and `?`, how many ways are there to replace the `?`s in S and T with `0` or `1` such that there exists a legal experiment with S as the initial state and T as the final state.

The experience in Riyadh left a lasting impression on Little Cyan Fish. Little Cyan Fish always felt like they had witnessed a miracle. After returning home, Little Cyan Fish found this problem very interesting and planned to include it in their competition next year to test the participants. Soon, the year was up. Please help Little Cyan Fish solve this problem. Of course, Little Cyan Fish understands that this number can be very large, so you only need to output the answer modulo 998 244 353.

Input

Each test case contains multiple sets of test data. The first line of input contains an integer T ($1 \le T \le 5\,000$), indicating the number of test data sets. For each test data set:

- The first line contains two positive integers n, k ($1 \le n \le 5000, 0 \le k \le n$).
- The next line contains a string S of length n composed of characters `01?`.
- The following line contains a string T of length n composed of characters `01?`.

It is guaranteed that the total sum of n across all test data does not exceed $5\,000$.

Output

For each test data set, output a number representing the answer.

Example

standard input	standard output
3 7 3 1111111 1100011 10 3 ?????????? ?????????? 5 2 ??1?1 ?0?1?	1 3384 3

Problem G. + and × with a sugar

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

此刻请将一切抛诸脑后
规矩你明白否

The background story in the Chinese statements is removed due to the translation difficulties. >_<

Little Cyan Fish has a sequence of n positive integers $a_1, a_2, \cdots, a_n$. Oscar Tang told Little Cyan Fish: "I want you to divide this sequence into several contiguous segments, where the profit of each segment is the product of all elements within that segment, and your task is to maximize the sum of all segment profits."

Little Cyan Fish found this problem quite challenging. Therefore, Little Cyan Fish hopes that you will provide the answer modulo $(10^9 + 7)$ and tell him the result.

Input

Each test case contains multiple sets of test data. The first line of input contains an integer T $(1 \le T \le 2 \times 10^5)$, indicating the number of test data sets.

For each test data set:

- The first line contains a positive integer n $(1 \le n \le 2 \times 10^5)$, which is the length of the sequence.
- The second line contains n positive integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$.

It is guaranteed that the total sum of n across all test data sets does not exceed 2×10^5.

Output

For each test data set, output a single line containing an integer, representing the answer requested by Little Blue Fish, taken modulo $(10^9 + 7)$.

Example

standard input	standard output
3 4 2 3 3 3 5 1 2 1 2 1 6 1 1 4 5 1 4	54 7 82

Problem H. Qingyu's Little Training Center

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

The background story in the Chinese statements is removed due to the translation difficulties. >_<

Little Cyan Fish has a training center on the continent of CP. In Little Cyan Fish's training center, there are m people. Little Cyan Fish has found n CP problems from his archive and hopes to select some problems from these to send to his m friends for training. However, to maximize the training effect, Little Cyan Fish wants to ensure that the problems he selects for each friend meet the following conditions:

- Each person should have just the right amount of training. Therefore, each friend receives exactly k problems for training.
- Different people should have different training methods. Therefore, any two friends must have received different sets of problems.

Of course, it is easy to see that the number of ways to choose k problems is $\binom{n}{k}$. Little Cyan Fish does not want many people solving the same problem—after all, he hopes that everyone's problem-solving abilities have a certain diversity! Therefore, if f_i represents how many people have received the i-th problem, the maximum value of f_i should be minimized as much as possible.

Formally, for each $1 \le i \le m$ and $1 \le j \le n$, if we denote $s_{i,j}$ as whether the i-th friend has received the j-th problem (if received, then the value is 1; otherwise, it is 0), then the above two conditions can be expressed as: s_i contains exactly k ones, and any two s_i and s_j are not equal ($1 \le i < j \le m$). Little Cyan Fish's goal is to minimize $\max_{1 \le j \le n} \left(\sum_{i=1}^{m} s_{i,j}\right)$.

Of course, merely finding the answer is far from enough—after all, Little Cyan Fish also needs to know how to operate specifically. Therefore, you also need to provide Little Cyan Fish with a valid scheme.

Input

Each test case contains multiple sets of test data. The first line of input contains an integer T ($1 \le T \le 10^5$), indicating the number of test data sets. For each test data set:

- The input consists of a single line containing three integers n, k, m ($1 \le n \le 20, 0 \le k \le n, 0 \le m \le \binom{n}{k}$), representing the number of problems in Little Cyan Fish's possession, the number of problems each person should solve, and the number of Little Cyan Fish's close friends.

It is guaranteed that the total sum of $\binom{n}{k}$ across all test data does not exceed 2^{20}.

Output

For each test data set, output in the following format:

- The first line outputs a number representing the answer, that is, the minimum possible value of the maximum f_i.
- Next, you need to describe the scheme that Little Cyan Fish should use. You should output m lines, each line being a binary string of length n, where the j-th number in the i-th line is 1 if the i-th person should receive the j-th problem.
- These m lines represent the strings $s_1, s_2, \cdots, s_m$ as described formally in the problem.

Example

standard input	standard output
3 4 1 3 5 2 3 10 10 1	1 0100 0001 1000 2 11000 10100 01100 1 1111111111

Problem I. Ineffable Cycle

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 1024 megabytes

撕掉标准人生的地图
就当我偏偏最喜欢迷路

The background story in the Chinese statements is removed due to the translation difficulties. $>\_<$

Little Cyan Fish falls into a dream. It seems to see its future life trajectory, represented as a simple undirected graph with n vertices and m edges. This graph has too many edges, too complex a structure, and too many cycles.

Little Cyan Fish decides to choose two vertices i and j, such that after removing these two vertices from the graph, the entire graph has no cycles.

The Little Cyan Fish is curious about how many ways there are to choose such a pair of vertices (i, j) $(1 \le i < j \le n)$.

Input

Each test case contains multiple sets of test data. The first line of input contains an integer T $(1 \le T \le 2 \times 10^5)$, indicating the number of test data sets. For each set of test data:

- The first line contains two integers n, m $(3 \le n \le 5\times10^5, 0 \le m \le \min(\frac{n(n-1)}{2}, 5\times10^5))$, representing the number of vertices and edges.
- The next m lines each contain two integers u, v $(1 \le u, v \le n, u \ne v)$, representing an undirected edge. It is guaranteed that there are no multiple edges or self-loops in the graph.

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

Output

For each set of test data, output a single line containing an integer, representing the answer.

Example

standard input	standard output
3 7 7 1 2 2 3 3 4 4 5 5 1 4 6 6 7 8 9 1 2 2 3 3 4 1 5 5 6 6 4 1 7 7 8 8 4 7 8 1 2 2 3 3 4 4 1 4 5 5 6 6 7 7 4	20 25 15

Problem J. Not a work of Idol

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

The background story in the Chinese statements is removed due to the translation difficulties. >_<

Little Cyan Fish wants to count the number of labeled d-regular graphs with n vertices. Little Cyan Fish wants to remind you that a d-regular graph is an undirected simple graph (with no multiple edges or self-loops) where each vertex has a degree exactly equal to d.

Two regular graphs G_1 and G_2 are said to be different if there exists a pair of vertices v_1 and v_2, such that $(v_1, v_2) \in E_1$ but $(v_1, v_2) \notin E_2$, or $(v_1, v_2) \in E_2$ but $(v_1, v_2) \notin E_1$.

Of course, this number can be very large, so he wants to take the result modulo a **small** prime number p that does not exceed 7.

Input

Each test case contains multiple sets of test data. The first line of input contains two integers T and p ($1 \le T \le 10^4, p \in \{2, 3, 5, 7\}$), representing the number of test data sets and the modulus.

The next T lines each contain two integers n, d ($0 \le d < n \le 10^{18}$).

Please note that, the modulus for all test cases in a single test file is the same

Output

For each set of test data, output a single line contains a single integer, representing the answer.

Example

standard input	standard output
5 3 3 2 9 6 20 10 100 10 1919810 114514	1 1 2 0 0

Problem K. Grotesque Team Reconstruction

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

The background story in the Chinese statements is removed due to the translation difficulties. $>\_<$

There exists a university with $n = 3k+1$ (with k being a non-negative integer) students who are passionate about programming competitions. Among these students, due to their shared interests and goals, there arises a strong desire to form teams. This desire can be abstracted as m given relationships: if the i-th relationship is (a_i, b_i), it indicates that student a_i wishes to team up with student b_i. In other words, this forms an undirected graph with n vertices (representing students) and m edges (representing team formation desires).

However, for some unknown reason, Little Dark Fish loves to break apart these dream-filled students in a grotesque manner, preventing them from forming standard teams of three to participate in important competitions. The evil plan is to forcibly assign these n students to two completely independent and isolated "new schools," which we will call school S_1 and school S_2. This assignment must strictly adhere to the following cruel and twisted rules:

- Each student must be assigned to one of the two schools and can only belong to one campus. That is, the intersection of the student sets S_1 and S_2 must be empty ($S_1 \cap S_2 = \varnothing$), and their union must encompass all students ($\{1, 2, \cdots, n\}$).
- The number of students assigned to the two schools, $|S_1|$ and $|S_2|$, must both satisfy the condition of being congruent to 2 modulo 3 (i.e., $|S_1| \equiv 2 \pmod 3$ and $|S_2| \equiv 2 \pmod 3$).
- Although the schools are separated, the students within each school must maintain some form of "unity." Specifically, considering only the students within school S_1 and the subgraph formed by their original team formation desires (i.e., the induced subgraph of S_1), this subgraph must be connected. Similarly, the induced subgraph of school S_2 must also be connected.

Little Dark Fish wants you to recall the definition of an induced subgraph:

- For an undirected graph $G = (V, E)$ and a subset of vertices $S \subseteq V$, the induced subgraph $G[S]$ is a new graph where the vertex set is S itself, and the edge set contains all edges from the original graph G where both endpoints are within the vertex set S. In other words, for any two distinct vertices $u, v \in S$, the edge (u, v) exists in the induced subgraph $G[S]$ if and only if the edge (u, v) exists in the original graph G.

Little Dark Fish wants you to determine whether there exists a valid team division scheme.

Input

Each test case contains multiple sets of test data. The first line contains an integer T ($1 \le T \le 10^5$), indicating the number of test cases. For each test case:

- The first line contains two integers n, m ($4 \le n \le 5 \times 10^5, 0 \le m \le 5 \times 10^5$), ensuring that $n \equiv 1 \pmod 3$.
- The next m lines each contain two integers u, v ($1 \le u, v \le n, u \ne v$), representing an edge.

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

Output

For each test case, if there exists a valid division scheme, output one line with `Yes`. Otherwise, output one line with `No`.

Example

standard input	standard output
6 4 4 1 2 2 3 3 4 4 1 4 3 1 2 1 3 1 4 4 4 1 2 1 3 1 4 2 3 4 6 1 2 1 2 1 2 1 2 3 4 3 4 4 0 7 6 1 2 1 3 2 4 2 5 3 6 3 7	Yes No Yes Yes No No

Note

In the first sample test case, Little Dark Fish only needs to assign students $\{1, 2\}$ to the first school and $\{3, 4\}$ to the second school to solve the problem.

In the second sample test case, we can find that no matter how the division is made, it cannot meet Little Dark Fish's conditions.

In the third sample test case, Little Dark Fish only needs to assign students $\{1, 4\}$ to the first school and $\{2, 3\}$ to the second school to solve the problem.

In the fourth sample test case, Little Dark Fish only needs to assign students $\{1, 2\}$ to the first school and $\{3, 4\}$ to the second school to solve the problem. Note that the problem does not guarantee that the undirected graph does not contain multiple edges, and this sample test case does contain multiple edges.

Problem L. Yearning for Yonder

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

This is an interactive problem.

The background story in the Chinese statements is removed due to the translation difficulties. >_<

Little Cyan Fish has a weighted tree of n verticies generated in the following way:

- First, generate a random labeled tree: uniformly randomly select from all n^{n-2} possible labeled trees.
- Then, independently assign random integer edge weights in the range $[1, K]$ to each edge, where K is a hidden parameter.

You cannot directly observe the structure of the tree or the edge weights, but Little Cyan Fish grants you a superpower: querying! Each time, you can query the distance between two vertices. Specifically, you can choose two vertices u, v ($1 \le u, v \le n$, $u \ne v$), and we will tell you the distance between these two vertices (i.e., the sum of the edge weights on the simple path connecting these two vertices).

Now, Little Cyan Fish wants you to determine all the edges and their weights within no more than $7n$ queries.

Interaction Protocol

Each test case contains multiple sets of test data. First, you need to read an integer T ($1 \le T \le 10^4$) indicating the number of data sets.

For each set of test data, you first need to read an integer n ($1 \le n \le 10^5$).

Next, the interaction process begins. You can make no more than $7n$ queries in each set of test data. To make a query, you need to output a line "`?` u v" ($1 \le u, v \le n$, $u \ne v$), describing a query. Then, you need to read the result from standard input.

To provide your answer, you need to output "`!` u_1 v_1 w_1 u_2 v_2 w_2 $\cdots$ u_{n-1} v_{n-1} w_{n-1}". You can output these edges in any order. The output of the answer will not count towards the $7n$ query limit. After you output the answer, you need to immediately read the next set of test data or terminate your program.

After outputting a query, **do not** forget to output a newline character and flush the output stream. To do this, you can use `fflush(stdout)` or `cout.flush()` in C++, `System.out.flush()` in Java, `flush(output)` in Pascal, or `stdout.flush()` in Python.

It is guaranteed that $1 \le K \le 10^4$, and the sum of all n in the test data does not exceed 10^5.

In this problem, it is guaranteed that the interaction library is **non-adaptive**. That is, the shape of the tree and the edge weights are determined before the interaction process. They will not change with your queries.

Example

standard input	standard output
2	
3	
	? 1 2
3	
	? 2 3
4	
	? 1 3
7	
	! 1 2 3 2 3 4
4	
	? 1 2
3	
	? 2 3
7	
	? 2 4
2	
	? 1 3
4	
	? 1 4
5	
	? 3 4
9	
	! 1 2 3 1 3 4 2 4 2

Problem M. Under the Epilogue

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 1024 megabytes

我仍然在
无人问津的阴雨霉湿之地
和着雨音
唱着没有听众的歌曲

The background story in the Chinese statements is removed due to the translation difficulties. >_ <

There are n positions, numbered from 1 to n. The i-th position has two parameters a_i and b_i ($a_i \le i \le b_i$). Position i can move to any positions j in the range $j \in [a_i, b_i]$.

Now, Little Cyan Fish wants you to make future decisions. You can choose any position u and direct Little Cyan Fish to start from u and take any number of steps, ultimately returning to u. Of course, you can also choose to take no steps at all and simply stay at u.

Starting from u and ending at u, let S represent the set of all positions that have been visited at least once. Little Cyan Fish asks you how many different sets S can be generated in total, modulo 998 244 353.

Input

The first line of input contains an integer n ($1 \le n \le 50$).

The next n lines each contain two integers a_i, b_i ($1 \le a_i \le i \le b_i \le n$).

Output

Output a single integer, representing the result modulo 998 244 353.

Example

standard input	standard output
5 1 2 2 5 3 4 2 5 4 5	12