Problem A. Today's Word

You have accepted an offer at VortaroEnMano Inc., a company committed to creating the most comprehensive Esperanto dictionary. Esperanto estas tre mojosa lingvo, so you work really hard to do your best – mostly to keep your job in the depression of employment.

Today, you're assigned to refactor the function called "Hodiaŭa Vorto", i.e. "Today's Word" in English. This word is generated from a string, denoted as S_k.

Here's how S_k is generated:

1. The process begins with an initial string, S_0, which is given. This string contains only lowercase English letters and has an even length.
2. For $n \geq 1$, S_n is generated in the following way: $S_n = S_{n-1}[0 \dots \frac{l}{2}-1] + S_{n-1} + \text{next}(S_{n-1}[\frac{l}{2} \dots l-1])$, where l is the length of S_{n-1} and $+$ is used to concatenate the strings. Note that the index of the string starts from 0.

The function $\text{next}(S)$ increments each character in the string S to the next letter in the alphabet, i.e., `a` changes to `b`, `b` to `c`, and so on, with `z` changing to `a`. For instance, $\text{next}(\texttt{abz}) = \texttt{bca}$.

Your task is to determine the suffix of $S_{10^{100}}$ with a length of m.

Input

The first line contains two integers n and m ($1 \leq n, m \leq 10^5$), representing the length of S_0 and the length of the desired suffix, respectively. It is guaranteed that n is an even number.

The second line contains a string, S_0, composed of n lowercase English letters.

Output

Output a string of length m, which represents the suffix you are tasked to determine.

Example

standard input	standard output
6 10 bocchi	wrwxrwxsxy

Note

In the provided example, $S_1 = \underline{\texttt{boc}}\ \underline{\texttt{bocchi}}\ \underline{\texttt{dij}}$.

Problem B. Honkai in TAIKULA

TAIKULA is a complex galaxy made up of n distinct stars, indexed from 0 to $n-1$ and interconnected through m directed star rails. Our tale unfolds with the daring crew of the Astral Express, specifically two members, March 7th and Dan Heng, assigned the critical task of delivering rare relics within TAIKULA. Their mission is disrupted by an unexpected onslaught from the Honkai at the Space Station. Fortunately, March and Dan, along with chief researcher Asta and security head Arlan, manage to repel the attackers.

However, these Honkai encounters have now become an unfortunate yet regular nuisance along the star rails, causing various losses. In response, Asta takes upon the task of estimating the potential losses associated with each star rail, assigning them integer values on his map. It is noteworthy that these integers can be negative since some of the weaker yet affluent Honkai might be defeated, leading to the Astral Express acquiring valuable equipment or materials.

The standard protocol for any business transportation within TAIKULA involves starting from a star x, traversing across other (at least one) stars, and eventually returning to the original star x for necessary maintenance. During this transportation, stars and star rails might be visited multiple times, including the starting star. Asta is keen on minimizing the incurred loss from each starting star as much as possible.

Interestingly, the Honkai exhibit a peculiar habit in TAIKULA. They only plunder odd-numbered goods during a transportation, and their aggression levels rise exponentially in the face of an even loss. This leads Asta to seek your expertise as a Trailblazer with the power of Stellaron. Your mission is to plan a transportation route from each starting star that results in the least possible odd loss. To expedite the process, you only need to calculate the minimum odd loss from each starting star.

Input

The first line contains two integers n, m ($1 \le n \le 1000$, $1 \le m \le 10^4$), indicating the number of stars and star rails.

Each of the following m lines contains 3 integers x_i, y_i, w_i ($0 \le x_i, y_i < n$, $|w_i| \le 10^7$), identifying a directed star rail from star x_i to star y_i with w_i loss.

Output

The output should consist of n lines. The i-th line should identify the minimum odd loss for the $(i-1)$-th starting star. If a transportation with an odd loss does not exist, output `Battle with the crazy Honkai`. Otherwise, if the minimum odd loss is finite, output the loss. If it is infinite, output `Haha, stupid Honkai`.

Examples

standard input	standard output

2 2
0 1 1
1 0 1
	Battle with the crazy Honkai
	Battle with the crazy Honkai

2 2
0 1 0
1 0 1
	1
	1

2 2
0 1 -1
1 0 0
	Haha, stupid Honkai
	Haha, stupid Honkai

2 2
0 1 2
1 0 -1
	1
	1

Problem C. GG and YY's Game

Two strategic players, GG and YY, are about to engage in a competitive game. The game takes place on an undirected graph, denoted as G. The graph is characterized by a parameter t ($t \geq 1$) and comprises n disjoint chains.

The rules of the game are as follows:

- GG and YY take turns to make a move on graph G, with GG always going first.
- During each turn, the player chooses one node, denoted as s, from the graph. Upon this selection, the player captures all nodes on the same chain as s that are at a distance of at most t from s, including s itself.
- Once a node is captured, it is eliminated from the graph, potentially causing the chain to break apart.
- The game proceeds until all vertices are captured, at which point the game concludes.

Both GG and YY aim to capture the maximum number of nodes. Let cnt_{GG} and cnt_{YY} represent the number of nodes captured by GG and YY, respectively. Assuming both players, GG and YY, employ their optimal strategies, your task is to determine:

- When $t = 1$, what is the value of $\text{cnt}_{\text{GG}} - \text{cnt}_{\text{YY}}$?
- When $t > 1$, which player wins the game?

Input

The first line contains one integer T ($1 \leq T \leq 100$), indicating the number of test cases.

The first line of each test case contains two integers n and t ($1 \leq n \leq 10^4$,$1 \leq t \leq 10^{18}$), indicating the number of chains and the parameter.

The second line of each test contains n integers $l_1, l_2, \ldots, l_n$, where each l_i ($1 \leq l_i \leq 10^{18}$) indicates that a distinct chain of length l_i exists in G.

Output

For each test, output the answer in a single line.

- When $t = 1$, output $\text{cnt}_{\text{GG}} - \text{cnt}_{\text{YY}}$.
- When $t > 1$, output `GG` if GG wins, or output `YY` if YY wins, or output `TIE` if the game results in a tie.

Example

standard input	standard output

2
2 1
1 5
2 2
1 5
	2
	GG

Problem D. Star Rail

In the far reaches of the universe, there is a special train called the Astral Express. The Astral Express, built by the renowned engineer, Aeon Akivili the Trailblazer, has the unique ability to travel throughout the galaxy using a network of star rails.

The universe is comprised of n stars, which can be conceptualized as n points on a 2D plane. The Astral Express begins its journey at a chosen star, referred to as the i-th star.

Pom-Pom, the conductor of the Astral Express, has an important task. Pom-Pom needs to draw a straight line that divides the remaining $n-1$ stars into two sets (possibly empty). This line must avoid intersecting any of the remaining $n-1$ stars, although passing through the i-th star is permissible. Pom-Pom will then choose one of these sets to visit, provided it contains exactly k stars. If there is no set that meets this condition, Pom-Pom will not select any. The number of different sets that can be selected is denoted as $A_{i,k}$.

Pom-Pom is curious about the matrix A formed by elements $A_{i,k}$. Your task is to help him determine this matrix. In other words, for each possible i $(1 \le i \le n)$ and k $(1 \le k \le n-1)$, you need to calculate the corresponding $A_{i,k}$, which represents the number of selectable sets when the Astral Express starts at the i-th star and the size of the set is k.

Input

The first line contains a single integer n $(2 \le n \le 2 \times 10^3)$, representing the total number of stars in the universe.

Each of the next n lines contains two integers x_i, y_i $(0 \le x_i, y_i \le 10^9)$, representing the coordinates of the i-th star in the universe. It is guaranteed that each star has a unique position.

Output

Output a matrix with n rows and $n-1$ columns, representing the matrix A. The element at the i-th row and the k-th column should denote $A_{i,k}$, which represents the number of selectable sets when the Astral Express starts at the i-th star and the size of set is k.

Examples

standard input	standard output

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

5
0 0
0 4
4 0
4 4
1 2
	4 4 4 1
	4 4 4 1
	3 6 3 1
	3 6 3 1
	4 4 4 1

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

Problem E. LCM Plus GCD

You are given two integers, x and k. Your task is to determine the total number of sets that contain exactly k distinct positive integers $a_1, a_2, ..., a_k$ such that the sum of their Least Common Multiple (LCM) and Greatest Common Divisor (GCD) equals x, i.e. $\mathrm{LCM}(a_1, a_2, \ldots, a_k) + \mathrm{GCD}(a_1, a_2, \ldots, a_k) = x$.

The LCM of a set of numbers is the smallest positive integer that each number in the set can divide, while the GCD is the largest positive integer that can divide each number in the set. For instance, $\mathrm{LCM}(4, 6, 8) = 24$ and $\mathrm{GCD}(4, 6, 8) = 2$.

As the result could potentially be a very large number, you should provide the output modulo $10^9 + 7$.

Input

One line contains two integers x and k ($1 \le x, k \le 10^9$).

Output

Output an integer denoting the answer modulo $10^9 + 7$.

Examples

standard input	standard output
14 2	3
14 3	4

Problem F. Timaeus

Timaeus, a budding alchemist in Mondstadt, has been tasked by his mentor, Albedo, to synthesize a specific type of alchemical product: Large Sweet Flowers. However, due to his limited skills, Timaeus can only create one Large Sweet Flower by combining B Regular Sweet Flowers in a single synthesis. Therefore, he has enlisted the assistance of two helpers: Sato and Mona.

Sato has a knack for increasing productivity, with a probability $P\%$ of producing twice the output during a single synthesis, which means to produce two Large Sweet Flowers by combining B Regular Sweet Flowers in a single synthesis. On the other hand, Mona has a knack for resource conservation, with a probability $Q\%$ of retrieving one Regular Sweet Flower during the process, which means to produce one Large Sweet Flower and simultaneously recover one Regular Sweet Flower by combining B Regular Sweet Flowers in a single synthesis. However, Timaeus can only choose one assistant for each synthesis.

Starting with a total of A Regular Sweet Flowers, Timaeus aims to maximize the expected number of Large Sweet Flowers he can produce. Therefore, he must optimally choose between assistants Sato and Mona for each synthesis. Please calculate the maximum expected number of Large Sweet Flowers he can produce.

Input

One line contains four integers A, B, P, Q ($1 \le B \le A \le 10^6$, $0 \le P, Q < 100$), representing the total number of Regular Sweet Flowers, the number of Regular Sweet Flowers needed to synthesize a Large Sweet Flower, the probability of producing double output, and the probability of retrieving a material, respectively.

Output

Output a floating-point number representing the maximum expected number of Large Sweet Flowers he can produce. Your answer will be considered correct if the absolute or relative error between the jury's answer and yours is less than or equal to 10^{-9}.

Examples

standard input	standard output
4 2 10 25	2.200000000000000
4 2 10 90	2.901000000000000

Problem G. Moving Boxes

In a large warehouse, there are n boxes lined up in a straight row. Each box i is initially placed at a location represented by the coordinate x_i. However, due to a reorganization plan, each box needs to be moved to a new position, y_i.

To assist with this reorganization, a specialized robot is at your disposal. This robot is designed to move at a speed of one unit of length per second and is capable of carrying only one box at any given time. The time taken by the robot to pick up and drop off boxes is negligible, but it requires C seconds to change its direction. Initially, you can set the robot in any position and facing any direction. But after all boxes are relocated, the robot must return to its original position and direction. The boxes do not need to be transported to the destination all at once, which means they can be dropped off halfway. Additionally, multiple boxes can coexist in the same location during the process.

Your challenge is to strategize the robot's movements such that the total time to relocate all the boxes is minimized. Please calculate minimum total time required for the robot to relocate all boxes.

Input

The first line contains two integers n and C ($1 \le n \le 10^5$, $1 \le C \le 10^9$), indicating the number of boxes and the time cost for changing direction.

Each of the following n lines contains two integers x_i and y_i ($1 \le x_i, y_i \le 10^9$, $x_i \ne y_i$), indicating the initial position and the end position of box i. It is guaranteed that all x_i are distinct from each other, and all y_i are distinct from each other.

Output

Output a single integer denoting the minimum time for the robot to relocate all boxes.

Examples

standard input	standard output

3 1
1 2
4 6
5 3
	12

3 4
5 10
9 1
8 6
	38

4 1
1 1001
1002 2
3 1003
1004 4
	4008

Note

Example 2: Start at position 1, heading in the positive direction. Move to position 5. Pick up box 1. Move to position 8. Drop off box 1. Pick up box 3. Change direction. Move to position 6. Drop off box 3. Change direction. Move to position 8. Pick up box 1. Move to position 10. Drop off box 1. Change direction. Move to position 9. Pick up box 2. Move to position 1. Drop off box 2. Change direction. The total time taken is 38 seconds.

Example 3: Start at position 3, heading in the positive direction. Pick up box 3. Move to position 1003.

Drop off box 3. Move to position 1004. Pick up box 4. Change direction. Move to position 1002. Drop off box 4. Pick up box 2. Move to position 2. Drop off box 2. Move to position 1. Pick up box 1. Change direction. Move to position 1001. Drop off box 1. Move to position 1002. Pick up box 4. Change direction. Move to position 4. Drop off box 4. Move to position 3. Change direction. The total time taken is 4008 seconds.

Problem H. Neil's Machine

Dr. Neil Watts, a Technician Specialist at Sigmund Corp., has developed a prototype machine that is capable of modifying the memories of terminally ill patients. He represents these memories using strings.

In this scenario, we use two strings of the same length, denoted as S and T. Both of these strings are composed of only lowercase English letters. The string S represents the patient's original memories, while the string T represents the target memories. The ultimate goal is to perform several operations on string S until it becomes identical to string T.

The memory modification machine performs operations using a basic function called **rshift k** where $1 \leq k \leq 25$. This function changes a letter to the k-th letter following it in the alphabet. In this context, the letter following `z` is considered to be `a`.

However, due to the potential for memory modifications to alter the course of the timeline, Dr. Watts can only apply the **rshift k** operation to a suffix of string S during each operation. The value of k and the length of the suffix can be arbitrarily chosen for each operation.

For instance, the string `uvwxyz` would be `uvzabc` if **rshift 3** is applied to the suffix beginning with `w`.

As excessive memory operations can lead to unintended consequences, Dr. Watts needs to minimize the number of operations performed. Therefore, he needs your help to determine the minimum number of operations required to transform string S into string T.

Input

The first line contains an integer n ($1 \leq n \leq 2 \cdot 10^5$), denoting the length of strings.

The second line contains a string S consisting of n lowercase English letters.

The third line contains a string T consisting of n lowercase English letters.

Output

Output an integer representing the minimum number of operations required to transform string S into string T.

Examples

standard input	standard output
4 aaaa aaaa	0
9 aaaaaaaaa aaabbbaaa	2
17 sofiawilumerrymee sofiawillumarryme	8
32 takethesebrokenwingandlearntofly takethesesunkeneyesandlearntosee	12

Problem I. Elevator

Imagine yourself in an elevator along with $n-1$ other people, making it a total of n people. Each person has chosen a floor they wish to alight at. When the elevator stops at any given floor, all those who have selected that floor will exit the elevator.

In this scenario, m unique floors have been selected. Your task is to determine the maximum possible number of people, including yourself, who could leave the elevator on the floor you have chosen.

Input

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

For each test case, only one line contains two integers n, m ($1 \le m \le n \le 10^4$), indicating the number of people in the elevator and the number of unique selected floors, respectively.

Output

For each test case, output one integer in a single line, indicating the maximum number of people who could leave the elevator at the floor you have chosen.

Example

standard input	standard output
1 6 3	4

Problem J. Similarity (Easy Version)

The easy version and the hard version are different problems, yet it is crucial to retain your solution code once it's accepted, as it might be useful later on.

Many place names bear similarities. Take, for instance, `Suzhou` and `Quzhou`. They are considered similar because they share a common substring `uzhou`. A substring is a contiguous sequence of characters within a string. For example, in the string `abcd`, `bc` is a substring of it, but `ac` is not.

We define the similarity between two strings as the length of their longest common substring. Thus, the similarity between `Suzhou` and `Quzhou` is 5, and between `Hangzhou` and `Chengdu` it is 2.

Given n place names, your task is to find two names that have the maximum similarity and output the similarity.

Input

The first line contains one integer T $(1 \le T \le 15)$, indicating the number of test cases.

For each test case, the first line contains an integer n $(2 \le n \le 50)$, indicating the number of places. Each of the following n lines contains a string s $(1 \le |s| \le 50)$, indicating the place names. The place names are guaranteed to consist only of lowercase English letters.

It is guaranteed that the sum of n over all test cases does not exceed 300.

Output

For each test case, output an integer in a single line representing the maximum similarity.

Example

standard input	standard output
2 2 jiangsu xiangtan 3 hangzhou chengdu wuxi	4 2

Problem K. Similarity (Hard Version)

The easy version and the hard version are different problems. However, it is advisable to get the easy version accepted first.

Many place names bear similarities. Take, for instance, `Suzhou` and `Quzhou`. They are considered similar because they share a common substring `uzhou`. A substring is a contiguous sequence of characters within a string. For example, in the string `abcd`, `bc` is a substring of it, but `ac` is not.

We define the similarity between two strings as the length of their longest common substring. Thus, the similarity between `Suzhou` and `Quzhou` is 5, and between `Hangzhou` and `Chengdu` it is 2.

Your task is to construct n **distinct** strings of the same length k, composed solely of lowercase English letters. Out of the $\frac{n(n-1)}{2}$ total pairings between these strings, the maximum similarity should be exactly m.

Input

One line contains three integers n, m, k ($2 \le n \le 300$, $0 \le m \le 50$, $1 \le k \le 100$).

Output

If there is no solution, output `No` in a single line.

Otherwise, output `Yes` in the first line. Then each of the next n lines contains a constructed string. Note that the strings should be pairwise distinct.

Example

standard input	standard output
2 4 8	Yes jiangsuu xiangtan

Problem L. Architect

Steve, a professional architect, is working on a project where he needs to assemble n cuboid building blocks to form a larger cuboid structure, with no overlapping among the small cuboids and no gaps left unfilled within the large cuboid. The lower left corner of the larger cuboid structure should be at $(0, 0, 0)$, and the upper right corner should be at (W, H, L). Before starting the actual assembly, Steve sketches an initial blueprint, placing the n cuboids in a specific arrangement. Your task is to verify whether this blueprint satisfies the aforementioned conditions.

Input

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 three integers W, H, L $(1 \leq W, H, L \leq 10^9)$, denoting the upper right corner of the larger cuboid structure.

The second line contains an integer n $(1 \leq n \leq 10^5)$, denoting the number of cuboid materials.

Each of the next n lines contains 6 integers x_l, y_l, z_l, x_r, y_r, z_r $(0 \leq x_l < x_r \leq W,\ 0 \leq y_l < y_r \leq H,\ 0 \leq z_l < z_r \leq L)$, denoting the coordinates of the cuboid, with its lower left at (x_l, y_l, z_l) and its upper right at (x_r, y_r, z_r).

It is guaranteed that the sum of n does not exceed 3×10^5 over all test cases.

Output

For each test case, if the proposed arrangement in the blueprint satisfies the given conditions, output `Yes` in a single line. Otherwise, output `No` in a single line.

Example

standard input	standard output

1
3 5 7
4
0 0 0 3 3 7
0 3 0 2 5 7
2 3 0 3 4 7
2 4 0 3 5 7
	Yes