The 2023 ICPC Vietnam

Southern Provincial Programming Contest

University of Science, VNU-HCM

October 15th, 2023

Contest length: **5 hours**

The problem set consists of 13 problems in 13 pages (*excluding the cover page*):

Problem A
A Game with Cows

Time Limit: 1 second
Memory Limit: 512 megabytes

Hieu and RR are two cowherd boys working in a barn. This barn has 10^9 stalls, numbered 1 to 10^9 from the left to the right. Each stall can hold at most one cow.

There are N cows numbered from 1 to N ($N \leq 10^6$). Every day at 3PM, the cows come back to the stalls from the lawn. Each cow will take a random stall. The i^{th} cow will take stall a_i and all a_i are pairwise distinct.

To make the feeding process easier, Hieu and RR want to move all N cows to the left (occupied stall 1 to stall N). They decided to make a game, taking alternative turns:

- Hieu is the first to move.
- In each turn, the player must move the rightmost cow (the cow in the stall $\max(a_i)$) to any empty stall on its left.
- If a player can not make a move, he loses.

Given the initial positions of N cows, your task is to determine who wins if they both play optimally.

Input

The input contains multiple test cases, starting with an integer T ($1 \leq T \leq 10^5$) – the number of test cases. The following lines contain the description of each test case.

Each test case starts with a single integer N – the number of cows. ($1 \leq N \leq 10^6$). The second line consists of N distinct integers $a_1, a_2, \dots, a_N$. ($1 \leq a_i \leq 10^9$).

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

Output

For each test case, output “`Hieu`” (without the quotes) if Hieu wins or “`RR`” (without the quotes) if RR wins, given that they both play optimally.

Sample Input	Sample Output
4 2 3 4 3 1 2 4 3 1 3 5 5 1 2 3 4 5	RR Hieu Hieu RR

Problem B
Bouquet

Time Limit: 1 second
Memory Limit: 512 megabytes

For the Vietnamese Women's Day in the next few days, Phuoc wish to buy a bouquet for his mother. He knows she holds a special affection for a lucky number, which has a value of n.

Upon visiting a flower stall, he orders a bouquet of exactly k flowers. A bouquet is constructed by combining different flowers in a particular sequence. There are precisely 10 varieties of flowers, each numbered from 0 to 9, and the availability of each type is deemed infinite. The product of the numbers on the selected flowers must match the lucky number n, which his mother cherishes.

The price of the bouquet is determined by the sequence in which Phuoc chooses to combine the flowers. For instance, selecting a type 8 flower followed by a type 3 flower results in a cost of 83. In contrast, choosing a type 3 flower and then a type 8 flower would make the price 38. When he chooses a type 4 flower followed by a type 6 flower, he will have a bouquet with the same lucky number $n = 24$, but the price will be 46.

Given his student budget, Phuoc aims to assemble a bouquet of k flowers at the lowest possible cost and the bouquet must match his mother's lucky number n.

Input

The input contains a single line of two integers n and k ($1 \le n \le 10^9; 1 \le k \le 10^4$).

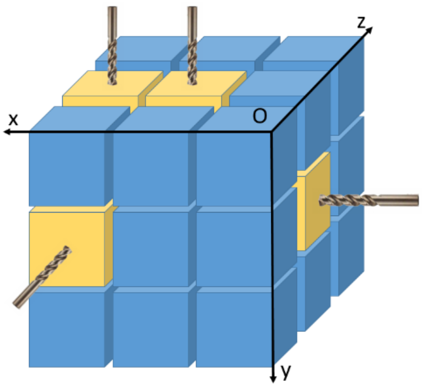
Output

The price of the bouquet. If there is no bouquet that satisfies the conditions, output -1.

Sample Input	Sample Output
12 2	26
34 2	-1

Problem C
Cube

Time Limit: 1 second
Memory Limit: 512 megabytes



Prof. Thanh designs a very special rectangular cube with dimensions of $n \times n \times n$ units. He builds a coordinate system O_{xyz} at a particular corner of the cube, with the three faces of the cube being the three planes O_{xy}, O_{yz}, and O_{zx}.

Each **unit block** of the cube is defined by a set of three numbers (x, y, z) as a point in 3-dimensional space $(0 \le x, y, z \le n - 1)$. Prof. Thanh says that **potential unit blocks** are located at (x, y, z), where **exactly one** of the coordinates is 0, which means that they are located on one of the three faces O_{xy}, O_{yz}, O_{zx}, but not on the axes.

Prof. Thanh chooses k **potential unit blocks** and for each of the blocks, he drills from the center of the square located on the cube's face and from the direction perpendicular to the face containing that cell. Suppose that he drills through the cube.

Your task is to help Prof. Thanh determine how many unit blocks are passed by at least one drill.

Input

The first line contains two integers, n and k $(1 \le n \le 10^9; 1 \le k \le 10^3)$.

In the following k lines, each line contains three integers x, y, z $(0 \le x, y, z \le n - 1)$ indicating the location of a potential unit block. For each line, exactly one of the three integers is 0.

Output

A single integer indicates the number of unit blocks passed by at least one drill.

Sample Input	Sample Output	Explanation
3 4 2 1 0 0 1 1 1 0 1 2 0 1	9	As illustrated in the figure above

Problem D
Danger Detection

Time Limit: 1 second
Memory Limit: 512 megabytes

A land is described by a grid of size $n \times n$ square units. The rows are numbered from 1 to n from top to bottom, and the columns are numbered from 1 to n from left to right. The cell located in row i and column j is called cell (i, j). Through an assessment, each cell has a specific load-bearing capacity, and cell (i, j) has a positive integer load-bearing capacity of a_{ij}.

A robot starts moving from cell $(1, 1)$ and needs to reach cell (n, n) by moving to adjacent cells (with a common edge). Suppose the robot has a weight of G, then entering a cell with a load-bearing capacity less than G is considered **dangerous**. Your task is to solve the following two problems:

1. Given the robot's weight G, find its path to minimize the number of dangerous cells it goes through. Determine the number of dangerous cells on such path.
2. For the given grid, determine the maximum weight of the robot so that it can move to the cell (n, n) without entering any dangerous cells.

Input

The first line contains a number 1 or 2 corresponding to problem 1 or 2 you need to solve.
If the first line contains the number 1, the second line contains two integers n and G. If the first line contains the number 2, the second line contains a single integer n. $(1 \le n \le 500; 1 \le G \le 5000)$.
In the next n lines, the i^{th} line contains n numbers on row i of the grid a with the value from 1 to 10^4.

Output

The output contains a single integer, which is the result of the corresponding problem.

Sample Input	Sample Output
1 5 6 7 1 3 4 6 6 2 1 7 5 2 6 7 2 3 4 2 1 1 7 9 1 6 1 6	3
2 5 7 1 3 4 6 6 2 1 7 5 2 6 7 2 3 4 2 1 1 7 9 1 6 1 6	2

Problem E
Exact Permutation Mapping

Time Limit: 1.5 seconds
Memory Limit: 512 megabytes

In a universe linked by cosmic tunnels, *Futuristic Interstellar Tubes (FIT)* beckon explorers for traveling between planets. To embark, travelers need to solve challenges related to permutations of positive integers.

Consider the set of permutations of $2n$ positive integers $1, 2, \dots, 2n$:

$$a_1, a_2, I, a_n, a_{n+1}, I, a_{2n}$$

with the following properties:

- The sequence a_1, a_2, I, a_n is in increasing order.
- The sequence a_{n+1}, I, a_{2n} is in increasing order.
- For all i $(1 \le i \le n)$, $a_i < a_{n+i}$.

All these permutations are sorted in lexicographic order and consecutively numbered starting from 1. Thus, each permutation has a serial number, and we call it the permutation index.

For example, with $n = 3$, we have all 5 permutations, after sorting the dictionary is numbered as illustrated in the table on the right.

Index	Permutation
1	1 2 3 4 5 6
2	1 2 4 3 5 6
3	1 2 5 3 4 6
4	1 3 4 2 5 6
5	1 3 5 2 4 6

Given the number n, your task is to solve the following two problems from the guardians of the Universal Passage:

1. Given an index k, find the permutation with index k.
2. Given a permutation $a_1, a_2, \dots, a_{2n}$, find the index of this permutation.

Input

The input contains at most 2000 test cases. Each test case is in one line, with a number 1 or 2 at the beginning of the line indicating the problem you need to solve.

- If the first number is 1, it is followed by two integers n and k $(1 \le n \le 1000; 1 \le k \le 10^9)$.
- If the first number is 2, it is followed by an integer n and $2n$ integers $a_1, a_2, \dots, a_{2n}$.

Note that the input is guaranteed correct, and the solution always exists.

Output

You need to output the results of each test case in the order they appear in the input.

Sample Input	Sample Output
1 3 4 2 3 1 3 5 2 4 6 1 4 2 2 4 1 2 3 6 4 5 7 8	1 3 4 2 5 6 5 1 2 3 5 4 6 7 8 3

Problem F
Fibonacci Power

Time Limit: 2 seconds
Memory Limit: 512 megabytes

In the heart of an ancient forest, adventurer **Minh the Sorcerer** embarked on a quest of solving enigmatic puzzles. A family heirloom journal unveiled a cryptic message: a hidden treasure guarded by mathematical riddles tied to the Fibonacci sequence.

To recap, Fibonacci numbers are defined as follows:

- $F(0) = 0$
- $F(1) = 1$
- $F(i) = F(i-2) + F(i-1)$ for all $i \geq 2$.

To unlock the treasure, Minh should quickly calculate the sum S of the first n Fibonacci numbers raised to the power of k.

Formally:

$$S = \sum_{i=1}^{n} F(i)^k$$

Given n and k $(1 \leq n \leq 10^{18}, 1 \leq k \leq 10^6)$, help Minh calculate S modulo 998244353.

Input

The input contains two integers n and k $(1 \leq n \leq 10^{18}, 1 \leq k \leq 10^6)$.

Output

You should print the value S modulo 998244353.

Sample Input	Sample Output	Explanation
3 2	6	$F(1)^2 + F(2)^2 + F(3)^2$ $= 1^2 + 1^2 + 2^2$ $= 6$
5 1	12	$F(1)^1 + F(2)^1 + F(3)^1 + F(4)^1 + F(5)^1$ $= 1^1 + 1^1 + 2^1 + 3^1 + 5^1$ $= 12$
1000 1000	954643773	

Problem G
Goals

Time Limit: 1 second
Memory Limit: 512 megabytes

Quang and Trong participate in a special soccer match. In this competition, each of them is given N soccer balls at random, each with a random weight in the range of $[1, 10^{18}]$.

The match unfolds as follows: the soccer field has N goals, numbered from 1 to N. Quang kicks off the game by shooting his N balls into these N goals, with one ball going into each goal (assuming both Quang and Trong have 100% shooting accuracy). He is free to choose the order in which the balls are shot. After Quang has taken all his shots, Trong considers the weight of Quang's balls in the current goals and then decides to shoot his N balls into the corresponding N goals, each ball into each goal.

At the end of the match, for the i^{th} goal, the player whose ball has a higher weight wins and gets i score ($i = 1, 2, \ldots, N$). If both balls in the i^{th} goal have the same weight, the score is split equally, meaning each player gets i/2 score. The player who has a higher total score wins the match.

As an example of the match, suppose $N = 3$, Quang is given N balls with weights 1, 2, and 4, and Thuc is given balls with weights 1, 3, and 5. Quang may shoot balls weighted 4, 2 and 1 on goals 1, 2 and 3, respectively. After seeing Quang's moves, Trong may shoot the balls weighted 3, 5 and 1 on goals 1, 2 and 3, respectively. In this case, Quang earns 1 point for slot 1, 0 points for slot 2, and $3/2 = 1.5$ points for slot 3. His total score is 2.5 points. The purpose of the above example is to illustrate the rules of the match. Quang and Trong may not shoot the balls as described because both play optimally.

Knowing that, during his turn, Trong is aware of the positions of Quang's balls. Trong chooses his shooting strategy to maximize the score he can earn. Meanwhile, during Quang's turn, he does not know the information about Trong's balls. Still, he is aware of Trong's optimal strategy and that the ball weights are uniformly and randomly distributed. Hence, Quang chooses his playing strategy to maximize the expected value of his total score.

Given N, please help Quang calculate the expected value of his total score at the end of the match if both players make their choices optimally as described.

Input

The input contains a single integer N ($1 \leq N \leq 100$), the number of goals.

Output

A single number indicating the expected value of Quang's total score at the end of the match. The result must have exactly six digits after decimal point, rounded if necessary.

Sample Input	Sample Output
1	0.500000
2	1.333333
100	598.874902

Problem H
Harry's Magical Number

Time Limit: 1 second
Memory Limit: 512 megabytes

In *Starlight Essence Laboratory (SELab)*, the famed wizard Harry Potter and his friend Hai are performing a magical show to students. Harry asks Hai to think of a magical number n. In Hai's mind, this number dances and sparkles. Harry does not know what it is, but he will try to guess!

1. First, Harry asks Hai to reveal a divisor A of n. Hai picks one and announces it.
2. Next, Hai will need to tell Harry a number B, but this number must not be a divisor of n.
3. Not stopping there, Harry then requests Hai to let him know a multiple C of n.
4. Lastly, Harry wishes for another number D, which must not be a multiple of n.

With a flourish of his wand and magical chimes, Harry will attempt to deduce the magical number n Hai thought of. However, Harry warns that sometimes there might be more than one solution, and he will try to identify the smallest magical number n. If it is undeterminable, Harry will humbly admit defeat and say "-1".

Now, standing on stage with Harry, help him determine the magical number n.

Note: To assist Harry, be aware that A, B, C, and D are all less than or equal to 10^9.

Input

The input consists of a line containing four integers A, B, C, and D ($1 \leq A, B, C, D \leq 10^9$).

Output

Your program must output a single line. If there is at least some number n for which A, B, C and D are significant, then that line must contain the smallest n possible. Otherwise, the line must contain -1.

Sample Input	Sample Output
2 12 8 2	4
3 4 60 105	6

Problem I
Intricate Polygons

Time Limit: 1 Second
Memory Limit: 512 megabytes

On a beautiful sunny day in Dam Sen Park, Mr. Don holds a bag of sticks of various lengths. Mr. Don has realized that, with these sticks, some sticks can be combined to create interesting polygons, while others cannot.

For example, Mr. Don has three sticks in his bag with lengths of 5, 3, and 4, respectively. He knows that with these three sticks, he can arrange them to form a beautiful triangle. However, the previous day, when Mr. Don had four sticks in his bag with lengths of 5, 3, 4, and 20, he knew there was no way to arrange them into a valid quadrilateral.

Here is your task: Help Mr. Don calculate how many different ways to choose sticks from his bag to create different polygons. Two polygons are considered different if at least one stick is used in one polygon but not present in the other, and vice versa.

Input

The first line contains an integer n $(1 \le n \le 5000)$ – number of sticks in Mr. Don's bag.

The second line contains n integers $l_1, l_2, \ldots, l_n$ $(1 \le l_i \le 5000)$, which are the lengths of the n corresponding sticks.

Output

The output contains a positive integer, which is the number of different polygons that can be created modulo $10^9 + 7$.

Sample Input	Sample Output	Explanation
4 2 2 2 2	5	There are 4 triangles with sides {2,2,2} and 1 quadrilateral with sides {2,2,2,2}
5 1 2 3 4 5	9	There are 9 polygons {2,3,4}, {2,4,5}, {3,4,5}, {1,2,3,4}, {1,2,3,5}, {1,2,4,5}, {1,3,4,5}, {2,3,4,5}, {1,2,3,4,5}

Problem J
Journey to Sequence Sum

Time Limit: 2 Seconds
Memory Limit: 512 megabytes

Once upon a time, in a realm where mathematics held great power, a young scholar named Quang discovered a mysterious sequence. This sequence, denoted as A_i, had a captivating formula:

$$A_i = i^K \times R^i.$$

Quang was determined to unlock its secrets.

As he delved deeper into his studies, he realized that his ultimate quest was to find the sum of the first n terms of this sequence $(A_1, A_2, \dots, A_n)$, which he called S_n.

To keep things manageable in this enchanted land, S_n had to be calculated modulo $10^9 + 7$. Can you help him compute S_n.

Input

The first line of input contains T $(1 \le T \le 10)$, the number of test cases.

Each test case consists of three lines, including K $(1 \le K \le 10^3)$, n $(1 \le n \le 10^{16})$, and R $(2 \le R \le 10^{16})$ respectively.

Output

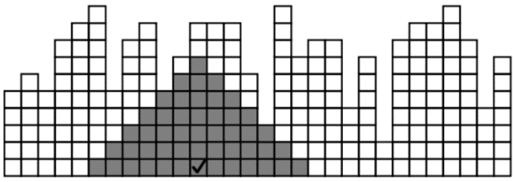
For each test case, print your answer in a line.

Sample Input	Sample Output
1 1 2 3	21

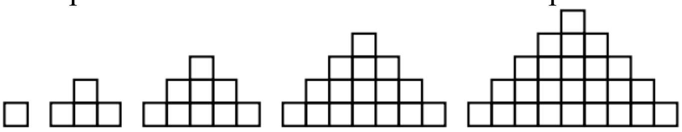
Problem K
Keystone Triangle

Time Limit: 1 Second
Memory Limit: 512 megabytes

In a classic art exhibition room, Leo Da Vi creates artwork from n consecutive stone pillars with respective heights $a_1, a_2, \ldots, a_n$. He wants to create an isosceles triangle sculpture, but currently, each stone pillar has a different height, similar to the illustrated figure beside.



Leo decided to start sculpting each stone pillar, always from the top to the bottom of each column, so that in the end, only an isosceles triangle remains from the stone pillars. He can only sculpt on each stone pillar and not transfer them to another pillar. The image below illustrates the first five stone triangle shapes that Leonardo desires, with heights of 1, 2, 3, 4, and 5, respectively.



Given the sequence of heights of each stone pillar, could you help Leonardo determine the maximum height the isosceles triangle can reach? In the illustrative figure above, with 30 pieces of stone, the tallest isosceles triangle has a height of 7.

Input

The first line of the input contains an integer n ($1 \leq n \leq 50000$), representing the number of pillars. The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq n$), indicating the heights of each pillar.

Output

Your program must produce a single line with an integer H, representing the maximum height a triangle could have at the end.

Sample Input	Sample Output
16 5 6 5 8 9 10 5 8 9 5 7 9 9 9 6 3	6
8 5 1 1 1 1 1 1 3	1

Problem L
Luffy and The One Piece

Time Limit: 1 second
Memory Limit: 512 megabytes

Captain Luffy, accompanied by his esteemed crew, is embarking on a quest to discover the legendary treasure known as "One Piece". In his possession, Captain Luffy holds the distinguished map of the Grand Line. This cartographic representation illustrates a maritime region, dimensioned as $m \times n$, subdivided into individual square units starting with an index of 1. Captain Luffy is currently stationed on an island positioned precisely at coordinates (x, y) on the said map. He was conferred a cryptic hint: "In the remotest location shall you seek, and by the k^{th} attempt, the treasure shall you meet."

Guided by this counsel, Captain Luffy strategizes his expeditionary endeavors as follows:

- For the inaugural voyage, he will navigate to the most distant part of the sea relative to the island and commence his exploration therein.
- During the subsequent sortie, Captain Luffy intends to scout the location that is second in remoteness to the island yet untouched by previous explorations.
- On the tertiary expedition, his objective is to explore the third most distant square from the island that remains uncharted.
- This methodical progression continues until the k^{th} exploration, where, concealed beneath the maritime abyss, the illustrious One Piece treasure is believed to reside.

Nonetheless, Captain Luffy, in his astuteness, recognizes the myriad potential locales that might harbor the treasure. Should he maintain this sequential trajectory, there's a looming concern that rival pirates might preemptively locate the treasure. So, Captain Luffy earnestly asks for your expertise. However, his primary requirement is to know the distance from the island to the presumptive treasure locale. With this knowledge, Captain Luffy is confident in leveraging his unparalleled intuition to pinpoint the precise resting place of the One Piece treasure. It is pertinent to note, as advised by Captain Luffy, that within this maritime expanse, all distances are gauged using the Manhattan metric.

For clarity, the Manhattan distance between two designated coordinates (x_1, y_1) and (x_2, y_2) is articulated as $|x_1 - x_2| + |y_1 - y_2|$.

Input

A single line contains 5 integers m, n, x, y, and k.

$(2 \le m, n \le 10^9; 1 \le x \le m;\ 1 \le y \le n; 1 \le k \le m \times n - 1)$

Output

A single number indicates the Manhattan distance from the island to the One Piece.

Sample Input	Sample Output
5 6 4 3 4	5
5 6 4 3 6	4

Problem M
Mingle Lineup

Time Limit: 3 seconds
Memory Limit: 512 megabytes

Imagine you are organizing a show at school. You already have n students from group A standing in a vertical line, with their heights in order as $a_1, a_2, \ldots, a_n$. Just then, a group of students B consisting of m students with heights $b_1, b_2, \ldots, b_m$ want to join the line.

The catch is that the students from group A want to keep their positions the same, but the students from group B are very flexible and can stand anywhere: right at the beginning, between any two students in the line, or at the end.

A **mistake** is counted when a taller student stands in front of a shorter one. So, after inserting the students from group B into group A, what is the minimum number of **mistakes** you can arrange?

Input

Each test consists of multiple test cases. The first line contains one integer t ($1 \leq t \leq 10^4$) — the number of test cases.

- The first line of each test case contains two integers n and m ($3 \leq n, m \leq 10^6$) - the number of students in group A and B, respectively.
- The second line each input test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^9$) – the heights of students in group A.
- The third line of each input test case contains m integers $b_1, b_2, \ldots, b_m$($1 \leq b_i \leq 10^9$) – the heights of students in group B.

It is guaranteed that the sum of n over all input data sets does not exceed 10^6 and the sum of m over all input data sets does not exceed 10^6.

Output

For each test case, output one integer — the minimum number of mistakes that you can arrange.

Sample Input	Sample Output
3 3 3 3 2 1 1 2 3 3 4 1 2 3 4 3 2 1 5 4 1 3 5 3 1 4 3 6 1	4 0 6