Problem A. Olympiad Participants

Time limit: 1 second
Memory limit: 512 megabytes

There are n students in the class.

Some of them participated in olympiads:

- a students participated in the mathematics olympiad,
- b students — in the computer science olympiad,
- c students — in the Russian language olympiad.

Some students may have participated in several olympiads at once.

Help the teacher estimate how many students might not have participated in any of these olympiads. Find the minimum and maximum possible number of students who did not participate in any olympiad.

Input

The first line of the input contains one number n — the number of students in the class ($1 \le n \le 2 \cdot 10^5$).

In the second, third, and fourth lines, the numbers a, b, and c are given respectively ($0 \le a, b, c \le n$).

Output

In the first line, output the minimum possible number of students who did not participate in any olympiad. In the second line, output the maximum possible number of such students.

Examples

standard input	standard output
10 7 5 3	0 3
7 7 7 7	0 0

Problem B. Suitable Envelope

Time limit: 1 second
Memory limit: 512 megabytes

Semyon Igorevich is the head of the programming club. The students from his club went to an off-site programming contest by the sea at the famous educational center “Cassiopeia”. Unfortunately, Semyon Igorevich could not attend the contest, and his students decided to send him greetings from the sea by sending beautiful paper postcards.

Each of the n students bought a postcard, each postcard being a rectangle with a height of h_i and a width of w_i. To optimize the sending of postcards, the students decided to buy one large envelope and put all the postcards in it one atop of the other. The envelope is a rectangle with a height of H and a width of W. A postcard is placed in the envelope such that the sides of the postcard are parallel to the sides of the envelope. A postcard can be rotated by $90°$. A postcard fits in the envelope if its sides do not exceed the corresponding parallel sides of the envelope.

The students want the purchased envelope to have the minimum area $H \cdot W$.

Determine the height and width of the envelope so that all postcards can be placed in it and its area is minimized.

Input

The first line contains an integer n, $1 \le n \le 10^5$ — the number of students. In the following n lines, the height and width of each postcard h_i and w_i are given, ($1 \le h_i, w_i \le 10^9$).

Output

Output two integers H and W — the height and width of the suitable envelope. If the sides are not equal, output the smaller side first.

Examples

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

Problem C. Merging Amulets

Time limit:	1 second
Memory limit:	512 megabytes

Hornet has acquired a lot of amulets, specifically a total of n! Each amulet is unique and possesses a special power. Let the power of the i-th amulet be denoted as a_i.

Since it is easy to get confused with a large number of amulets, she can merge a non-empty subsegment of amulets into one more powerful amulet once. Suppose she chooses the segment $[i, j]$. Then the power of the new amulet will be equal to $\texttt{lcm}(a_i, a_{i+1}, \ldots, a_j)$, where lcm denotes the least common multiple of the numbers $a_i, a_{i+1}, \ldots, a_j$. Thus, after merging the amulets from i to j, she will have amulets with powers $a_1, \ldots, a_{i-1}, a_{j+1}, \ldots, a_n, k$, where k denotes the power of the resulting amulet.

Since Hornet is interested in the final combat power, she wants to recalculate it after merging the amulets. The combat power of a set of amulets is defined as the GCD of all the amulets in the set, where GCD or gcd denotes the greatest common divisor.

Hornet is interested in the sum of the combat power values for all possible ways to merge the amulets in the segment into one more powerful amulet. Formally, let $f(i, j)$ denote the combat power of Hornet's amulets that will result from merging the amulets from the i-th to the j-th inclusive. It will be equal to $\texttt{gcd}(a_1, \ldots, a_{i-1}, a_{j+1}, \ldots, a_n, \texttt{lcm}(a_i, \ldots, a_j))$. Hornet wants to calculate $\sum_{i=1}^{n} \sum_{j=i}^{n} f(i, j)$.

Help her calculate this value.

Input

The first line contains an integer n — the initial number of amulets ($1 \le n \le 2 \cdot 10^5$).

The second line contains n integers a_i — the initial powers of the amulets that Hornet possesses ($1 \le a_i \le 10^7$).

Output

Output a single number — the answer to the problem. Since the answer may be large, output it modulo 998 244 353.

Examples

standard input	standard output
5 2 6 9 3 6	44
6 1 2 3 4 5 6	85

Problem D. Building a Stage

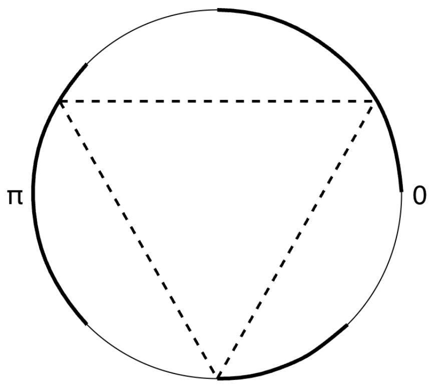
Time limit: 1 second
Memory limit: 512 megabytes

To award the winners of the olympiad, it has been decided to build a new stage. The award ceremony will take place in a circus, so the stage must be a triangle that lies inside an arena — a circle of radius r. The bases of the stage are the vertices of the given triangle.

On the circumference — the boundary of the circle — three non-intersecting arcs are designated. It has been decided that exactly one base of the stage should lie on each arc.

The organizers want to make the stage as large as possible in area, and you must help them by determining what the maximum area can be.

Below is an example of the stage for the test cases from the example:



In the figure, the designated arcs are shown with bold lines, and the optimal stage is shown with dashed lines.

Input

The first line contains a single real number r — the radius of the circle ($1 \leq r \leq 100$).

The second line describes the arcs. Fix an arbitrary radius vector of the given circle, and we will consider its direction as zero. Each arc is described by two real numbers: the angles in radians by which the given radius vector must be rotated counterclockwise so that its end points to the ends of the arc.

Thus, six real numbers $a_1, a_2, b_1, b_2, c_1, c_2$ are given — the angles that define the ends of the three given arcs ($0 \leq a_1 < a_2 < b_1 < b_2 < c_1 < c_2 \leq 2 \cdot \pi$).

Output

You need to output a single real number — the maximum area of the stage.

The answer must be output with an absolute or relative error of at least 10^{-6}. In other words, if the correct answer is a, and your output is b, the condition $\frac{|a-b|}{\max(|a|,1)} \leq 10^{-6}$ must hold.

Example

standard input
5 0.0 1.570796 2.356194 3.926991 4.712389 5.497787

standard output
32.4759526419

Problem E. Cubic Bushes

Time limit: 1 second
Memory limit: 512 megabytes

Boy Petya often walks with his mother in the park "Dekartgof.Dekartgof"is a classic park known for its strict shapes and straight lines.

One of the attractions of "Dekartgof"is the Alley of Cubic Bushes. A cubic bush is a plant made up of several identical cubic segments stacked on top of each other. There are a total of n bushes planted in a row in the Alley of Cubic Bushes. Petya is passionate about geometry, so he particularly enjoys walking through this alley. On his first visit to the alley, Petya recorded the height of each bush. The height of the bushes is the number of cubic segments in it.

In the Alley of Cubic Bushes, Petya especially likes symmetry, so he looks for sequences of bushes whose heights form a stepped palindrome. A stepped palindrome is a sequence of odd length where, in the first half, the height of each subsequent bush increases by one compared to the previous one, with the tallest bush in the center of the sequence, and then the heights of the bushes decrease by one until the end of the sequence. The heights of the bushes at equal distances from the center are the same. For example, $[1, 2, 3, 2, 1]$ is a stepped palindrome, while $[1, 2]$ and $[1, 5, 2]$ are not. Among all stepped palindromes, Petya wants to find the longest one.

Petya visits the park every month. The gardeners in "Dekartgof"try to trim the cubic bushes so that the alley always looks the same, but Petya noticed that during each visit to the park, one of the bushes changes its height, so the length of the longest stepped palindrome may change.

Petya is still young and cannot solve the problem, so help him find the maximum length of the stepped palindrome during his first visit to the park and in all subsequent visits.

Input

The first line contains one number n — the number of bushes in the alley ($1 \le n \le 10^5$).

The second line contains n numbers a_i — the initial heights of the bushes ($1 \le a_i \le 10^7$).

The third line contains one number q — the number of weeks when Petya visited the park ($1 \le q \le 10^5$).

In the following q lines, there are two numbers i x — the height of bush i has become x ($1 \le i \le n$, $1 \le x \le 10^7$).

Output

Output $q + 1$ lines — the maximum length among the stepped palindromes during Petya's first visit to the park and in all subsequent visits.

Example

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

Problem F. A Mess Nobody Needs

Time limit: 1 second
Memory limit: 512 megabytes

Once, while cleaning up the mess in his room, Petya and Varya found n threads and a pair of scissors. Naturally, to get rid of them, they invented a game. Petya goes first, and after that, the players take turns.

A turn consists of one action: the player chooses a thread of length x. After that, they must cut it into two threads of natural lengths a and b, such that $\gcd(a, b) > 1$, and return these threads back into the game. Here, $\gcd(x, y)$ denotes the greatest common divisor of the numbers x and y.

Note that after each turn, the number of threads in the game increases by one. The player who cannot make a move loses.

Petya is your good friend, and he wants to win against Varya in this game at all costs. Help him determine if he can win regardless of his opponent's moves.

Input

Each test consists of several test cases. The first line contains a single integer t — the number of test cases ($1 \le t \le 5000$). The following describes the test cases.

The first line of each test case contains an integer n — the initial number of threads in the game ($1 \le n \le 10^5$).

The second line of each test case contains n integers s_i — the lengths of the threads ($2 \le s_i \le 10^{12}$).

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

Output

For each test case, output "`Yes`" on a separate line if Petya can defeat Varya, and "`No`" otherwise.

Example

standard input	standard output
6 1 7 1 38 6 34 11 17 34 17 11 5 2 3 4 5 6 5 7 8 9 10 11 7 12 15 16 21 25 27 49	No Yes No Yes Yes No

Problem G. The Magic Suitcase

Time limit: 1 second
Memory limit: 512 megabytes

The sorceress Sofia received a new suitcase as a birthday gift. Sofia's suitcase is equipped with a super-modern electric combination lock. The lock on the suitcase is a square with a side length of n, where each cell of the lock contains a unique number from 1 to n^2.

Sofia devises her code—a specific arrangement of numbers in the cells of the lock that only she will know. After that, she will rearrange the numbers in a different order, and an intruder who does not know her code will not be able to break the lock. Sofia wants her code to be very secure.

Sofia knows that wizards and sorcerers will try to guess the code to break into her suitcase, so she wants the code for the lock to be an *anti-magic square*—an arrangement of numbers such that all $2n + 2$ sums: the sums of the numbers in the rows, columns, and the two main diagonals, are distinct.

Sofia is currently very busy with spells and potions, so help her come up with such a code that represents an anti-magic square, or tell her that it is not possible to create such a code.

Input

The first line of input contains a single integer n, $1 \leq n \leq 1000$—the side length of the square.

Output

If the code exists, output "`Yes`" on the first line. Then output n lines with n numbers in each line—the desired code. If multiple codes are possible, any of them can be output.

If such a code cannot be devised, output "`No`".

Examples

standard input	standard output
3	Yes 9 7 3 4 2 5 8 1 6
1	No

Problem H. Submask Parity

Time limit: 1 second
Memory limit: 512 megabytes

Andrey gave Kirill a gift — an array of integers $a_1, a_2, \ldots, a_n$, where $0 \le a_i < 2^{30}$.

Kirill is studying bitwise operations and is trying to understand how they can be used to determine whether a number is even or odd.

During his study, he noticed that for some numbers, the result of the bitwise `AND` operation turns out to be *special* — for example, when applying this operation to a number with another number leaves it unchanged.

Andrey told Kirill that an integer y is called a *submask* of a number x if

$$y \,\&\, x = y,$$

where "&" denotes the bitwise `AND` operation. In binary representation, this means that after padding both numbers with leading zeros to make them the same length, in all positions where y has a 1, x also has a 1. In other words, y can be obtained from x by replacing some of the 1-bits of x with zeros.

For example, 1 is a submask of 3, since $1 = 2^0$, $3 = 2^0 + 2^1$. However, 2 is not a submask of 12, since $2 = 2^1$, while $12 = 2^3 + 2^2$.

Kirill became curious: does there exist an integer x ($0 \le x < 2^{30}$) such that the number of elements in the array that are submasks of x is **odd**? The number x itself does not have to be present in the array.

If such a number exists, find any one of them. If no such number exists, report that there is no solution.

If the same number appears multiple times in the array, it is counted as many times as it occurs.

Input

The first line contains a single integer t — the number of test cases ($1 \le t \le 10^4$).

The descriptions of the test cases follow.

Each test case begins with a line containing a single integer n ($1 \le n \le 10^5$) — the number of elements in the array.

The next line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i < 2^{30}$).

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

Output

For each test case, print -1 if there does not exist an integer x ($0 \le x < 2^{30}$) such that the array a contains an odd number of its submasks.

Otherwise, print any such x.

Example

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

Problem I. Playlist

Time limit: 1 second
Memory limit: 512 megabytes

Katya listens to music. There are already several songs in the player's queue. The title and length of each song are known.

The player plays the song that is first in the queue and immediately removes it from there. Sometimes Katya comes across a new track that she wants to listen to next, at which point she presses the "Play Next" button. This song becomes the first in the player's queue, but the current song continues to play until the end. If this happens when the current song has finished, the next song to play will be the new track.

Your task is to determine the order in which Katya will listen to the songs and when each of them will start.

Input

The first line contains an integer n — the number of songs in the player at the moment when Katya started listening to music ($1 \le n \le 100\,000$).

The next n lines each contain two values: $name_i$, len_i, where $name_i$ — the title of the song, and len_i — the length of the song in seconds ($1 \le len_i \le 10^9$).

The following line contains an integer m — the number of events adding songs to the front of the queue ($0 \le m \le 100\,000$).

The next m lines each contain three values: t_j, $name_j$, len_j, where t_j — the moment in time the song is added in seconds ($0 \le t_j \le 10^9$), $name_j$ — the name of the added song, and len_j — its length in seconds ($1 \le len_j \le 10^9$).

It is guaranteed that the times of addition are given in non-decreasing order. The titles of the songs consist only of Latin letters, and their length does not exceed 20.

Output

For each played song, output one line containing two values: $name_k$ and $start_k$, where $name_k$ — the name of the song, and $start_k$ — the moment in time when this song started playing. The songs should be listed in the order in which Katya listened to them.

Example

standard input	standard output
3 Abracadabra 223 Pedro 145 Believer 220 3 223 Abracadabra 223 223 STAY 120 1024 Friday 1234	Abracadabra 0 STAY 223 Abracadabra 343 Pedro 566 Believer 711 Friday 1024

Problem J. Quantum Cats

Time limit: 1 second
Memory limit: 512 megabytes

Alyona is an expert in quantum physics. One day she decided to generalize the famous experiment with a cat and a box to n cats and n boxes. To do this, Alyona needs to conduct n experiments. For convenience, she numbered all the cats, all the boxes, and all the experiments with numbers from 1 to n.

In each experiment, there must be exactly one cat in each box. To avoid any inaccuracies related to some cats' biases towards certain boxes, each cat must visit each box exactly once over all n experiments. Moreover, for additional control, Alyona plans to monitor one of the boxes and the corresponding cat in it during each experiment. In the i-th experiment, she intends to observe the i-th box and wants the i-th cat to be in it at that moment, to ensure that over all n experiments, Alyona will observe each box and each cat exactly once.

Help Alyona find a possible arrangement of cats in boxes for each experiment, if it exists.

Input

The input consists of one natural number n ($1 \le n \le 300$) — the number of cats, boxes, and experiments.

Output

If the arrangement exists, output a table of size n by n, where the value in the i-th row and j-th column will contain the number of the cat that should be in the j-th box during the i-th experiment.

If the arrangement does not exist, output the integer -1.

Examples

standard input	standard output
1	1
2	-1
3	1 3 2 3 2 1 2 1 3

Note

No cats were harmed during the experiments.

Problem K. Restoring Weights

Time limit: 3 seconds
Memory limit: 512 megabytes

In the village of graphs, there lived two researchers, Lena and Misha. They found a connected undirected graph consisting of n vertices and m edges.

Some edges had known positive weights, while others had unknown weights. Lena and Misha decided to determine how many ways there are to assign integer values from 1 to l inclusive to the edges with unknown weights so that, for each vertex v, the shortest path from vertex 1 to vertex v has length exactly d_v.

Since the number of possible assignments can be very large, the answer should be given modulo $10^9 + 7$.

Input

Each test contains one or more test cases. The first line contains a single integer t ($1 \le t \le 50$) — the number of test cases.

The descriptions of the test cases follow.

The first line of each test case contains three integers n, m, and l ($2 \le n \le 50$; $n - 1 \le m \le \frac{n(n-1)}{2}$; $1 \le l \le 10^9$) — the number of vertices, the number of edges, and the maximum possible weight of an unknown edge.

The next m lines describe the edges of the graph. Each edge is given by three integers u_i, v_i, w_i ($1 \le u_i, v_i \le n$; $u_i \ne v_i$; $-1 \le w_i \le l$; $w_i \ne 0$) — the endpoints of the edge and its weight. If $w_i = -1$, the weight of this edge is unknown and can be chosen from 1 to l. If $w_i \ne -1$, this is the known weight of the edge.

The last line of each test case contains n integers $d_1, d_2, \ldots, d_n$ ($0 \le d_v \le 10^{12}$), where d_v is the required length of the shortest path from vertex 1 to vertex v.

It is guaranteed that the graph is connected and contains no loops or multiple edges.

Output

For each test case, output a single integer — the number of ways to assign weights to the edges with unknown weights from 1 to l such that, for every vertex v, the sum of the edge weights on the shortest path from vertex 1 to vertex v equals d_v, taken modulo $10^9 + 7$.

Example

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

Problem L. Long Jump Home

Time limit: 2 seconds
Memory limit: 512 megabytes

Few people know, but in fact, rabbits do not jump high right after birth! Everything comes at a cost — therefore, rabbits buy their jumping abilities at the rabbit store. So, the rabbit Xeni came to this store, which is located at point 0 on the number line.

In the store, there are n abilities, the i-th of which costs c_i carrots and teaches the rabbit to jump a distance of x_i along the number line. That is, after purchasing the i-th ability, Xeni will be able to jump a distance of x_i in either direction at any moment in time.

Immediately after her purchases at the store, Xeni wants to jump to point L ($L \neq 0$), where her home is located. Help her figure out the minimum number of carrots she will need to spend to have the ability to reach point L with the jumps she has acquired, or tell her that it is impossible.

Input

Each test consists of several sets of input data. The first line contains a single integer t — the number of sets of input data ($1 \leq t \leq 1\,000$). The description of the sets of input data follows.

The first line of each set of input data contains two integers n and L — the number of abilities in the store and the coordinates of Xeni's home ($1 \leq n \leq 3\,000$; $|L| \leq 3\,000$; $L \neq 0$).

The second line of each set of input data contains n numbers x_i — the lengths of the jumps that can be acquired ($1 \leq x_i \leq 3\,000$).

The third line of each set of input data contains n numbers c_i — the costs of the abilities ($1 \leq c_i \leq 10^9$).

It is guaranteed that the sum of n across all sets of input data does not exceed $3\,000$.

Output

For each test, output in a single line the minimum number of carrots Xeni will need to spend to jump home, or -1 if it is impossible.

Example

standard input	standard output
5 3 5 2 4 6 3 1 2 2 555 5 9 7 9 3 12 6 8 14 100 3 5 4 13 6 10 15 13 4 3 2 100 4 -21 7 7 9 4 5 1 10 10	-1 7 8 9 1

Problem M. Remainder of the Sum of Remainders

Time limit: 1 second
Memory limit: 512 megabytes

Given an integer n. You need to compute the sum $i \bmod j$ for all pairs (i, j) such that $1 \leq i, j \leq n$.

In other words, you need to compute:

$$S = \sum_{i=1}^{n} \sum_{j=1}^{n} (i \bmod j)$$

.

The operation $a \bmod b$ means the remainder of the division of a by b.

Since the sum can be large, compute the answer modulo $998\,244\,353$.

Input

The first line of input contains the number n ($1 \leq n \leq 10^{12}$).

Output

Output a single number — the answer to the problem modulo $998\,244\,353$.

Example

standard input	standard output
5	26

Note

Consider the example.

Let's build a table of remainders $i \bmod j$ for $i, j = 1 \ldots 5$:

$i \backslash j$	1	2	3	4	5
1	0	1	1	1	1
2	0	0	2	2	2
3	0	1	0	3	3
4	0	0	1	0	4
5	0	1	2	1	0

Summing the numbers in the table gives us 26.