Problem A. Farmer

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

You are given three numbers n, k and x. Construct an array A with the following properties:

- A contains exactly n elements.
- All elements of A are positive integers smaller than 10^6.
- There are **exactly** k pairs of indices (i, j) such that: $1 \le i < j \le n$ and $|A_i - A_j| \ge x$.

Input

Single line of input contains three numbers n ($1 \le n \le 1000$), k ($0 \le k \le \frac{n \cdot (n-1)}{2}$), x ($2 \le x \le 1000$).

Output

In single line of output print n integers, elements of array A. In case there is no array with needed properties, print -1

Examples

standard input	standard output
3 2 5	1 8 2

Problem B. MST Camera

Input file:	*standard input*
Output file:	*standard output*
Time limit:	7 seconds
Memory limit:	512 mebibytes

On a recent team contest your team was the best, and therefore won the best award. A camera. But it's not an ordinary camera, manufacturer of the camera, the famous "MST" company claims that this camera has unique capability. If you use it to picture some set of undirected edges of some graph, it is capable of calculating wheter it is possible to form a tree with these edges, and even better, if edges are weighted it is capable of finding a tree with minimum possible sum of weights of edges.

Your task is to check whether your camera works or not. You have a matrix with R rows and C columns, as well as the number N - the number of nodes in the graph. In every field of the matrix, you have one undirected edge of the graph. You should answer on Q queries, where each query is some submatrix of the original matrix. Answer to that query is the sum of weights of edges of minimum spanning tree, formed with edges in the given submatrix.

Formally, in field which is in row i and column j ($1 \le i \le R$, $1 \le j \le C$), you have three numbers $U_{i,j}$, $V_{i,j}$ and $W_{i,j}$, which means that in the field (i, j), there is an edge between node $U_{i,j}$ and $V_{i,j}$, with weight $W_{i,j}$. After that you have Q queries. Each query is described by four numbers X_1, Y_1, X_2, Y_2 ($1 \le X_1 \le X_2 \le R$, $1 \le Y_1 \le Y_2 \le C$), where (X_1, Y_1) is the upper left corner of the given submatrix, and (X_2, Y_2) is the bottom right corner of the submatrix. For each query consider graph with all N nodes and edges from the given submatrix. If there exists minimum spanning tree, you should print the sum of weights of tree edges. If spanning tree doesn't exist, you should print "-1" (quotes for clarity). For each query, condition: $\frac{2}{3} \le \frac{X_2-X_1+1}{Y_2-Y_1+1} \le \frac{3}{2}$ holds.

Input

In first line, there are numbers N, R, C and Q ($2 \le N \le 40$, $1 \le R, C \le 250$, $1 \le Q \le 200000$).

R rows follow and in each of them $3C$ numbers. In ith row the numbers are: $U_{i,1}$, $V_{i,1}$, $W_{i,1}$, $U_{i,2}$, $V_{i,2}$, $W_{i,2}$,...,$U_{i,C}$, $V_{i,C}$, $W_{i,C}$ ($0 \le W_{i,j} \le 65535$, for each $1 \le j \le C$).

Q rows follow and in each of them 4 numbers - X_1, Y_1, X_2 and Y_2.

Output

Print Q rows, in ith row answer to the ith query.

Examples

standard input	standard output
4 3 4 3 1 2 1 1 2 2 1 2 3 1 2 100 2 3 2 2 3 3 2 3 1 2 3 101 3 4 3 3 4 1 3 4 2 3 4 102 1 1 3 4 1 2 3 3 3 4 3 4	3 4 -1

Problem C. Winning Ballot

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Loznica is a city in Serbia, famous by history, culture, pleasant weather... and lottery! Lottery in Loznica is held with following rules:

- Ballot contains combination of N natural numbers, smaller than 10^{18} will be picked.
- In this lottery, numbers can repeat and order of them is important.

Aljoha, the hero of our story, using some obscure utilities, managed to find out some information about the next winning ballot. Let the future combination be L_i, $1 \le i \le N$. Aloha managed to find out an array containing $N-1$ numbers, in which the ith number A_i represents the largest number which divides both L_i and L_{i+1}.

Now Aljoha wants to bet, and for that noble goal he needs help. Print one combination which satisfies the constraints or -1, if that kind of combination doesn't exist. If there are more combinations that satisfy given constraints, print any of them. Note that only combinations in which all numbers are strictly smaller than 10^{18} are valid.

Input

First line contains number N ($1 \le N \le 10^5$), length of combination.

Second line contains $N-1$ positive integers not greater than 10^9, describing the information that Aljoha found out.

Output

Print N numbers, strictly less than 10^{18}, describing some combination which satisfies the constraints or -1 if there is no such combination.

Examples

standard input	standard output
4 3 4 10	3 12 20 10
4 3 4 6	-1

Problem G. Rats

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

You are given an infinite line covered with a periodically repeating string A (there are infinitely many concatenated copies of the string A in the line). **The line doesn't have a beginning or an end.** You are given a set S with M strings. You need to build new string B as concatanation of strings from S. String B must satisfy the following conditions:

- After covering a new empty infinite line with infinitely many concatenations of the string B, the line should be identical with string A.
- In case there are several valid strings B or several valid constructions of the string B, you should choose B and its construction which minimize number of strings used from S.

You can use the same string from S several times, but every time you count it as new string. You can concatenate all strings in any order, but you are not allowed to change the order of letters in the string. In case there is no proper way to build some string B, print -1.

Input

- The first line contains string A ($1 \le |A| \le 500$).
- The second line contains the integer M ($1 \le M \le 10^5$), number of strings in set S.
- Each of next M lines contains one string from S, i-th line contains string L_i ($1 \le |L_i| \le |A|$). Sum of lengths of strings from set S is smaller than 10^6 ($\sum_{i=1}^{M} |L_i| \le 10^6$).

Output

Print one integer — minimum number of string instances from S needed to build string B.

Example

standard input	standard output
baabaa 3 a b c	3

Note

You can use one string "b" and two strings "a", to build $B =$ "aba":

...baabaabaabaa...

.....abaabaabaaba...

Problem H. Coins

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	128 mebibytes

You are given an $N \times N$ matrix and in exactly N of its cells there is a single coin. You are going to play a game on this matrix where in one turn you can take a coin and move it to any adjacent cell. Two cells are adjacent if they share a side. However, during these moves, no two coins may occupy the same cell at the same time. Your goal is to make every row and column contain exactly one coin in as few moves as possible. Determine the minimum number of moves required.

Input

In the first line of input is the number N ($N \leq 200000$): the number of rows, columns and coins.

In the $(i+1)$-th line are two integers r_i and c_i, denoting the initial row and column of the i-th coin. It is guaranteed that all pairs (r_i, c_i) are different.

Output

Print a single integer: the minimum number of moves required to win at the game.

Example

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

Problem I. Marbles

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	128 mebibytes

You are given a track of length L. On it there are N marbles at positions $A_1, A_2 \cdots, A_N$ and N switches at positions $B_1, B_2, \cdots, B_N$, both of negligably small size. In the beginning, you direct each marble left or right and they all start moving at the constant speed of 1 in the given direction. When two marbles collide, they bounce off each other elastically, meaning that they continue moving in opposite directions at the same speed. If a marble collides with the beginning or end of the track it bounces off with the same speed in the opposite direction. Keep in mind that it takes exactly one second for a marble at position i to travel to the position $i + 1$ or $i - 1$, and it also takes exactly one second to travel from 1 back to 1 while changing directions, or to travel from L back to L while changing directions. Note that marbles never stop on the switches.

The goal is to have a marble on top of every switch, i.e. to have all marbles simultaneously on the corresponding switches.

You need to find the minimum amount of time needed to achieve this.

Input

In the first line of input you are given L ($L \leq 10^9$) and N ($N \leq 3000$)

In the second line of input you are given $A_1, A_2 \cdots, A_N$, the initial positions of the marbles. All A_i are guaranteed to be distinct, and $1 \leq A_i \leq L$.

In the second line of input you are given $B_1, B_2 \cdots, B_N$, the initial positions of the marbles. All B_i are guaranteed to be distinct, and $1 \leq B_i \leq L$.

Output

Print one integer: the minimum amount of time to have a marble on top of every switch. If solution does not exist, print -1.

Example

standard input	standard output
4 2 2 4 1 4	1

Problem J. Notebook

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	128 mebibytes

Ivan is writing down numbers in his notebook. In the beginning he has a set of integers S written down. Afterwards, he may write down new numbers in his notebook, by using the following operations:

- If he has the number x written, he may write down $2x$.
- If he has the number x written, and x is divisible by 2, he may write down $\frac{x}{2}$.
- If he has the **distinct** numbers x and y written, he may write down $x \text{xor} y$.

Denote by $f(S)$ the minimal number Ivan can write down in his notebook for the starting set S.

You are given an array of length N and Q queries where you have to perfortm one of the following operations:

- Change the value of the $a[x]$ to y.
- Find the value of $f(\{a[L], a[L+1], \cdots, a[R]\})$.

Input

In the first line of input is the numbers N ($N \leq 100000$): the length of the array.

In the second line of input is N integers $a[1], a[2], \cdots, a[N]$ ($0 < a[i] < 2^{62}$), the elements of a.

In the thirst line of input is the number Q ($Q \leq 100000$): the number of queries.

The following Q lines describe the queires. A query can either be of the format "$1xy$" meaning set $a[x]$ ($1 \leq x \leq N$) to y ($0 < y < 2^62$), or of the format "$2lr$" meaning find the value of $f(\{a[L], a[L+1], \cdots, a[R]\})$ ($1 \leq L \leq R \leq N$).

Output

For every query of type two print the value of $f(\{a[L], a[L+1], \cdots, a[R]\})$ in a single line.

Example

standard input	standard output
3 3 5 15 3 2 1 3 1 2 11 2 1 2	3 1

Problem K. Aunts

Input file:	*standard input*
Output file:	*standard output*
Time limit:	7 seconds
Memory limit:	128 mebibytes

The organizing comittee of this contest is made from a bunch of slackers who's rather do anything than prepare a contest. As a result of this, N of them reported that they have to bring some medicine to their aunt, so that they can get away without doing anything. All of their aunts live within a $A \times B$ matrix. They all report their absence to Dzoni, who takes a note of the cell that their aunt's house is located as well as the altitude it's located in. Dzoni doesn't know the actual altitudes, but he knows that in every two adjacent cells in the matrix, the absolute difference of their altitudes is **exaclty** 1. Having this in mind, after every claim made by a member of the organizing comittee he tries to reconstruct all the altitudes, meaning that he will check if its possible to assign each cell with an altitude so that all of the reports are true. If after some report, Dzoni is unable to reconstruct the altitudes (and since it's Dzoni, if it's possible, he'll be able to do it), it's obvious that the organizing comittee member that just made the report said something really stupid. Your job is to find the member who said something really stupid.

Input

In the first line of the input there are three integers N, A and B denoting the number of comittee members ($N \leq 10^5$), the number of rows and the number of columns of the matrix ($A, B \leq 10^9$).

In each of the following N lines, there are three integers R_i, C_i and H_i denoting the row, column ($R_i \leq A, C_i \leq B$) and altitude ($1 \leq H_i \leq 10^8$) of the i-th organizing comittee's alleged aunt's house.

Output

In the first and only line write the index of the comittee member who said something really stupid (comittee members are indexed from 1). If such a member doesn't exist, print "bravo komisijo" (meaning "congratulations, ogranizing comittee" in Serbian) without the quotation marks.

Example

standard input	standard output
4 3 5 1 2 1 2 1 3 2 5 21 2 2 8	3

Problem M. Anagrams

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

Two words are called ¡¡anagrams¿¿ if they contain the same letters but in a different order. For example `word` and `drow` are anagrams, but `word` and `worm` are not. In this problem you will be supplied with a set of words. All you have to do is to pick out the word with the most anagrams and report how many anagrams it has.

Input

Input begins with a number n, which is the number of words in the set ($0 \le n \le 1000$). Set contains n words, each on a single line. All words consist of lower case letters only. No word will contain more than 6 letters. It is guaranteed that set will contain at least one word that has an anagram in the list.

Output

Output consists of one word — the word with the most anagrams within the set, followed by a space, followed by the number of anagrams in set (excluding word itself). The word displayed will be the first occurrence in the input file of the anagram. If more than one word has the same highest number of anagrams, display only the one that comes first in the input file.

Example

standard input	standard output
6 nat cat act out tac ant	cat 2
6 worm word galo drown goal drow	word 1

Problem N. Banking

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

The bankers of Somebank have collected day-to-day data of the daily profits (and losses) of the shares they hold. Based on these numbers they decide to calculate which days would have been the most profitable to buy and sell their shares, so they can compare that with their actual gain.

Your task is to write a program that computes the optimal ¡¡run¿¿ in the sequence of numbers, the subsequence that maximizes the profit. The subsequence you compute is represented by the 1-based indexes of the first and last numbers in the subsequence. We are asking for exactly one date to buy and one date to sell shares since otherwise the solution would be simple: keep your shares on dates that the profit is non-negative.

Input

Input file has the following format:

- One line with a single integer N, $(1 \le N \le 10^6)$ — the length of the sequence to follow.
- One line with N integers p_i $(-1000 \le p_i \le 1000)$ — the profit (or loss) on date i.

The integers are separated by single spaces. At least one of the integers is positive.

Output

Output file contains a single line with two integers k and l $(1 \le k \le l \le N)$, such that the sum of the k-th until the k-th integer is maximized, boundaries included. When k and l can be selected by different ways, choose minimal k, then minimal l.

Example

standard input	standard output
11 -3 1 -1 2 3 1 -1 2 -3 -5 7	2 8
9 1 -2 3 -1 -1 3 -2 2 -4	3 6

Problem O. Dimensions

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

Teacher has given his pupils a table of widths, heights, lengths and volumes of some cuboid. Each row on the table has one of the 4 values missing; the pupils — and your program — have to work out the missing value and write it in the table such that the values on each line represent the width, height, length and volume of one cuboid.

Input

Input is a series of lines, each containing 4 integers: w, h, l and v representing the width, height, length and volume of a cuboid in that order. The integers are separated by a single space. $0 < l, w, h < 100, 0 < v < 10^5$. In each row, one of the values has been replaced by a zero. The final row contains 0 0 0 0 and should not be processed.

Output

Output is the same series of lines but with the zero in each line replaced by the correct value for length, width, height or volume as appropriate. It is guaranteed that the new value is always an integer.

Example

standard input	standard output
1 0 2 6 5 5 5 0 0 2 2 80 8 0 9 576 0 0 0 0	1 3 2 6 5 5 5 125 20 2 2 80 8 9 9 576

Problem P. Product of Digits

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

Given positive integer N, calculate product of digits for each positive integer up to N and report the greatest product.

Input

Input file contains one integer N ($1 \le N \le 2 \cdot 10^9$).

Output

Output file should contain greatest product for given N.

Examples

standard input	standard output
1	1
101090000	43046721
28994	10368
4876	2268
2789	1008