Problem A. And Xor Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

You are given a tree of n nodes. Each node has a non-negative integer value v_i.

Each path (from vertex i to vertex j) on the tree has an and-value A_{ij} which is the bitwise-and of the values of all nodes on the path. Similarly, each path has an or-value O_{ij} and an xor-value X_{ij} corresponding to the bitwise-or and bitwise-xor of the values of the nodes on the path respectively.

Compute the following three values:

$$\sum_{i,j} A_{ij}^2, \sum_{i,j} O_{ij}^2, \sum_{i,j} X_{ij}^2$$

where each sum ranges over all n^2 paths on the tree.

As the answers may be large, report each sum modulo 998244353.

Input

The first line of input contains a single integer n ($1 \le n \le 10^5$) — the number of nodes in the tree.

The second line of input contains n integers $v_1, v_2, \ldots, v_n$ ($0 \le v < 2^{25}$) — the value of each node in the tree.

The following $n - 1$ lines each contain two integers a_i, b_i — the endpoints of edge i.

Output

Output 3 integers — the sum of squares of all and-values, or-values, and xor-values respectively. Each sum should be reported modulo 998244353.

Examples

standard input	standard output
2 14 2 2 1	208 592 488
5 3 9 14 7 12 4 1 4 3 4 5 3 2	769 4627 1697
12 10 3 8 13 6 2 3 14 1 5 10 6 10 1 6 2 2 10 9 7 2 9 9 11 3 7 8 2 5 7 4 7 12 2	825 20705 12035

Problem B. Balanced Permutations

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	256 megabytes

Given a permutation p of size n, let a (contiguous) subarray of p be 'unstable' if the maximum value contained within the subarray is its first or last element. A permutation is considered 'balanced' if it has the minimum number of 'unstable' subarrays over all permutations of size n.

Given integers n, l, and k, report the l-th lexicographically-minimum 'balanced' permutation and the k-th lexicographically-maximum 'balanced' permutation of size n. If no such permutation exists output -1 instead.

Input

The only line of input contains three integers n, l, and k ($1 \leq n \leq 10^5$, $1 \leq l, k \leq 10^{18}$) — the length of the desired permutation and the indices of which lex-min and lex-max permutation should be provided.

Output

Output two lines. The first line should contain the l-th lexicographically-minimum 'balanced' permutation of size n — denoted p.

The second line should contain the k-th lexicographically-maximum 'balanced' permutation of size n — denoted q.

p and q should satisfy $1 \leq p_i, q_i \leq n$ for all $1 \leq i \leq n$.

If p or q does not exist (ie. there do not exist l or k 'balanced' permutations of size n) then report an answer of -1 instead.

Examples

standard input	standard output
3 1 2	1 3 2 1 3 2
4 9 13	3 1 4 2 -1

Problem C. Cyclic Shifts

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

We are given a permutation p of the integers 1 to n.

In a given operation we can choose $k > 0$ indices $1 \le x_1 < x_2 < \cdots < x_k \le n$ and cyclic shift the corresponding indices of the permutation one to the right.

$$p_{x_2} := p_{x_1}, p_{x_3} := p_{x_2}, p_{x_4} := p_{x_3}, \ldots, p_{x_k} := p_{x_{k-1}}, p_{x_1} := p_{x_k}$$

Applying this operation for a given k costs $\frac{1}{k}^{\star}$ dollars.

Your goal is to sort the given array using at most 2 dollars.

* For the purposes of the grader, the exact cost will be computed as $10^{-8} \left\lceil \frac{10^8}{k} \right\rceil$.

Input

The first line of input consists of a single integer n ($1 \le n \le 5 \cdot 10^3$).

The second line of each test case contains n integers $p_1, p_2 \ldots, p_m$ ($1 \le p_i \le n$) — the permutation to sort. It is guaranteed that the p_i will form a permutation.

Output

The first line of output contains a single integer m — the number of operations you used.

Then follow m lines of output.

The $i+1$-st line of output contains a binary string of length n s_i. If the j-th character of s_i is 1, then j is an index in the i-th cyclic shift (and the reverse if the j-th character is 0).

Each of the m lines of output must all contain at least one '1'.

Examples

standard input	standard output
3 2 1 3	4 011 110 111 011
4 1 2 3 4	0

Problem D. Distinct Subsequences

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	256 megabytes

Ethan has a binary string s of length n. He wants to give Justin a subsequence of s as a present for his birthday. To make the present special, he wants to make sure the length of the subsequence is exactly Justin's favorite number — k.

Compute the number of distinct presents Ethan could give to Justin. As this value may be large, compute the answer modulo 998244353.

Note: in this problem 'distinct' refers to the value of the subsequence. If a potential present appears as a subsequence of s in multiple locations it is counted exactly once.

Input

The first line of input contains two integers n and k ($1 \le k \le n \le 2 \cdot 10^5$) — the length of string s and the length of the desired subsequence respectively.

The second line of input contains a binary string s of length n.

Output

Output the number of distinct subsequences of s of length k, modulo 998244353.

Examples

standard input	standard output
5 3 00110	5
12 8 000111000000	12
31 12 1110100111110110101111010100010	3985

Problem E. Epidemic Escape

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	256 megabytes

There is a spaceship located at the origin of a 2-dimensional plane. There are n infection points in the plane.

At time $t = 0$ each infection will start spreading, creating a circle of radius t centered around the infection point. Simultaneously, at time $t = 0$ the spaceship will also start moving with speed 1, and with some shield level k_j. The spaceship will continue moving in a straight line until a moment comes when the number of infection circles that contain it is at least its shield value (at which point it will be consumed by the combined infection).

The spaceship's captain wants us to consider q scenarios for how to steer the ship. In each potential scenario, the spaceship will move in the direction of the point (x'_j, y'_j) with shield value k_j. Note that the given point is only to indicate direction, and the spaceship will continue moving even after passing (x'_j, y'_j).

For each scenario, compute the time at which the spaceship will be destroyed, or report that the spaceship will survive forever.

Input

The first line of input contains a single integer n ($1 \le n \le 10^5$).

The next n lines of input contain two integers x_i y_i, ($|x_i|, |y_i| \le 10^8$).

The $n + 2$-nd line of input contains a single integer — q ($1 \le q \le 10^5$).

The next q lines of input contain three integers — x'_j, y'_j, k_j,($|x'_j|, |y'_j| \le 10^8$, $1 \le k_j \le 5$).

Output

The output contains q lines of output — the time of destruction for each of the captain's plans (Your answer will be accepted if absolute or relative error is at most 10^{-6}).

If the spaceship will survive forever, output -1 instead.

The case $x'_j = y'_j = 0$ should have been an invalid test-case. On such queries, output -1.

Examples

standard input	standard output
5 5 -3 5 4 -6 2 -5 0 4 1 2 -3 -10 1 6 -9 1	8.700255424092125 3.2260195622572536
8 4 -1 4 -8 0 9 4 -7 -5 -2 5 -5 7 5 -9 2 10 4 -8 1 7 -7 5 -10 8 2 -9 9 2 4 -7 5 -1 -10 2 6 -3 2 2 -9 3 -10 -10 1 5 9 1	3.1677629681247024 26.162950903902267 5.461488320163311 6.363961030678928 -1 5.2894082216425735 3.726779962499649 4.6097722286464435 2.9294423792014115 4.7617289402064875

Problem F. Five Letter Warning

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	256 megabytes

Zack is concerned about his upcoming programming contest. In particular, he's concerned that the contestants might have seen some problems from a math contest that he had previously organized. It would be simple to ask the new contestants whether they had competed in his other contest, but that would give away too much information to the contestants! Instead, he has come up with the following devious plan.

First, Zack will ask a contestant for a string s. He will then compute all five-letter subsequences of s and count how many of them are palindromic. If a certain string appears multiple times as a five-letter subsequence of s, he will count it once for each time it appears. If too many subsequences are palindromic, he will ban that contestant from competing, otherwise he will let them through.

Unfortunately, Zack has been having issues with executing this plan for one s that has been given to him. Can you count the five-letter palindromic subsequences of s for him?

As this value may be large, compute the answer modulo M.

Input

The first line of input contains the string s ($1 \leq |s| \leq 10^6$). You may assume that s only contains characters with ASCII codes between 33 and 126 inclusive (all such characters are printable, non-whitespace characters).

This is followed by a line containing a single integer M ($2 \leq M \leq 10^9$) — the modulo with which to compute your answer.

Output

Output a single integer — the number of five-letter palindromic subsequences of s, modulo M.

Examples

standard input	standard output
cmimccmimc 998244353	34
GoodLuckToday!!!(~.^)IHopeYouHaveFun!! 311	69
aaaaaa 1000000000	6

Problem G. Gridlandia

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

The continent of Gridlandia is a squares of side length n, divided into n^2 square contries of side length 1. Each country has the resources to choose at most one of its four sides and build a 'wall' covering that entire side. However, since all the countries are at war, no two countries are willing to have their walls touch, even at the wall's endpoints.

Find the maximum number of walls that can be built in Gridlandia, and construct a configuration where the maximum number of walls are built.

Input

The only line of input contains a single integer n ($1 \le n \le 10^3$) – the number of squares on each side of the grid.

Output

The output should consist of n lines.

Each line should contain n characters, each being blank (represented by '.') or one of 'UDLR'.

A dot represents a country with no walls, whereas each of the characters 'UDLR' represents a country that builds a wall on its 'Up', 'Down', 'Left', and 'Right' sides respectively.

Examples

standard input	standard output
1	U
2	LU DR

Problem H. Holiday Regifting

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

In a town there live n people. Each person has some capacity c_i for how many gifts they can store in their house.

There also exist m friendships in the town. Each friendship has a 'mentor' corresponding to the friend in the friendship with the higher index.

Initially, no member of the town has any gifts. Every day, Santa comes to town and tasks an elf with giving a single gift to the member of the town with index 1. Unfortunately, this elf has a lot of work ahead of them.

If the elf gives a present to a town member whose house will not be filled by the present, they will accept it. However, if the town member's house would be completely filled by the present they will throw out all of their gifts (emptying their house) and tell the elf to deliver one gift to each of that town member's mentors, in increasing order of index. Any remaining leftover gifts will be thrown in the town incinerator.

The elf wonders what it will do if a town member has more mentors than gifts to give, but luckily all town members have at least as much capacity in their house as the number of mentors they have.

However, there is some additional complexity regarding how the elf gives out gifts. While the elf is in the middle of completing a gift-giving order, they can try to deliver a gift to a different nearly full house, causing a nested request. In this case, the elf will always deliver gifts corresponding to the most recent request (keeping track of a 'call stack' of requests to be carried out). It can be shown that the elf's actions throughout the day will be completed in a finite number of operations.

Santa, noticing the town's gift-giving shenanigans, is worried that on Christmas Day there will be no remaining gifts in the town, as they will all have been thrown in the incinerator. He has tasked you with finding the first day on which there are no gifts remaining in any house in the town. If this day will never come, output -1 instead.

In the positive case, as the answer may be large, compute the first such day modulo 998244353.

Note: throughout the gift-giving exercise, no house will ever be at full capacity for gifts. Any person will throw out gifts exactly when they would reach capacity when given the incoming gift.

Input

The first line of input contains two integers n, m ($1 \le n \le 10^4$, $0 \le m \le 3 \cdot 10^4$) — the number of people in the town and the number of friendships in the town respectively.

The second line of input contains $c_1, c_2, \ldots, c_n$ ($2 \le c_i \le 10^5$) — the capacity of each person's house.

The following m lines of input contain two integers u_i, v_i ($1 \le u_i < v_i \le n$) — the people in the i-th friendship. It is guaranteed that all listed friendships will be distinct.

Output

Output the first time when all houses in the town contain zero gifts. As the time may be large, output its value modulo 998244353. If no such time exists, output -1.

Examples

standard input	standard output
5 10 4 3 2 2 2 1 3 3 4 1 4 1 5 2 5 2 4 1 2 2 3 3 5 4 5	24
3 0 95 13 77	95
12 14 6 7 17 16 20 14 17 16 6 11 6 14 4 11 3 5 2 5 9 11 7 12 1 3 5 9 3 7 3 8 4 9 1 9 4 7 2 11 1 12	8739360

Problem I. Julienne the Deck

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Daniel is trying to shuffle his deck of n distinct cards. To do this, he has created a new shuffle operation which he calls 'Julienning' the deck.

First he chooses an integer i, $1 \leq i < n$. He then simultaneously reverses the first i cards of the deck, and the last $n - i$ cards of the deck.

For example, if his deck is initially in the permutation $p = [1, 4, 3, 2, 5, 6]$, then after a Julienne operation with $i = 4$ his deck becomes $p' = [2, 3, 4, 1, 6, 5]$.

If Daniel starts with a sorted deck of n cards, how many permutations of his deck can he achieve using any number (possibly zero) of Julienne operations?

As the answer may be large, output the value modulo 998244353.

Input

The first line of input consists of single integer n ($1 \leq n \leq 10^{12}$) — the number of cards in the deck.

Output

Output a single integer — the number of achievable permutations, modulo 998244353.

Examples

standard input	standard output
1	1
1000000000000	516560941
3	6

Problem J. Knight's Tour Redux

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Consider a n by n chessboard with squares labelled $(1, 1)$ through (n, n). On this chessboard lies a long knight. The long knight can move from square (x, y) to (x', y') if one of the two following conditions hold:

- $|x - x'| = 3$ and $|y - y'| = 1$
- $|x - x'| = 1$ and $|y - y'| = 3$

In essence, it is a normal chess knight, but longer.

A 'tour' of the chessboard is a sequence of squares $S_1, S_2, S_3, \ldots S_n$ such that for all $1 \leq i \leq n - 1$ the move from S_i to S_{i+1} is a valid move for a long knight. Such a tour is considered 'complete' if and only if the tour visits each row and column of the chessboard **exactly** once.

For each positive integer n, determine whether it is possible for a complete tour of an n by n chessboard to exist and construct one such tour, if possible.

Input

The first line of input consists of a single integer n ($1 \leq n \leq 10^5$) — the size of the chessboard.

Output

If it is not possible to create a 'complete' tour output the string "IMPOSSIBLE" in the only line.

Otherwise, output "POSSIBLE" in the first line.

The next n lines should contain the values x_i, y_i — the position of the i-th square in the complete tour.

Examples

standard input	standard output
1	POSSIBLE 1 1
2	IMPOSSIBLE