# Codeforces Round #365 (Div. 2)

## A. Mishka and Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Mishka is a little polar bear. As known, little bears loves spending their free time playing dice for chocolates. Once in a wonderful sunny morning, walking around blocks of ice, Mishka met her friend Chris, and they started playing the game.

Rules of the game are very simple: at first number of rounds $n$ is defined. In every round each of the players throws a cubical dice with distinct numbers from $1$ to $6$ written on its faces. Player, whose value after throwing the dice is greater, wins the round. In case if player dice values are equal, no one of them is a winner.

In average, player, who won most of the rounds, is the winner of the game. In case if two players won the same number of rounds, the result of the game is draw.

Mishka is still very little and can't count wins and losses, so she asked you to watch their game and determine its result. Please help her!

### Input

The first line of the input contains single integer $n$ $n$ $(1 \le n \le 100)$ — the number of game rounds.

The next $n$ lines contains rounds description. $i$-th of them contains pair of integers $m_i$ and $c_i$ $(1 \le m_i,\ c_i \le 6)$ — values on dice upper face after Mishka's and Chris' throws in $i$-th round respectively.

### Output

If Mishka is the winner of the game, print "Mishka" (without quotes) in the only line.

If Chris is the winner of the game, print "Chris" (without quotes) in the only line.

If the result of the game is draw, print "Friendship is magic!^^" (without quotes) in the only line.

### Examples

| input |
| --- |
| 3<br>3 5<br>2 1<br>4 2 |
| output |
| Mishka |

| input |
| --- |
| 2<br>6 1<br>1 6 |
| output |
| Friendship is magic!^^ |

| input |
| --- |
| 3<br>1 5<br>3 3<br>2 2 |
| output |
| Chris |

### Note

In the first sample case Mishka loses the first round, but wins second and third rounds and thus she is the winner of the game.

In the second sample case Mishka wins the first round, Chris wins the second round, and the game ends with draw with score 1:1.

In the third sample case Chris wins the first round, but there is no winner of the next two rounds. The winner of the game is Chris.

# B. Mishka and trip

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little Mishka is a great traveller and she visited many countries. After thinking about where to travel this time, she chose XXX — beautiful, but little-known northern country.

Here are some interesting facts about XXX:

1. XXX consists of $n$ cities, $k$ of whose (just imagine!) are capital cities.
2. All of cities in the country are beautiful, but each is beautiful in its own way. Beauty value of $i$-th city equals to $c_i$.
3. All the cities are consecutively connected by the roads, including $1$-st and $n$-th city, forming a cyclic route $1 — 2 — \ldots — n — 1$. Formally, for every $1 \le i < n$ there is a road between $i$-th and $i + 1$-th city, and another one between $1$-st and $n$-th city.
4. Each capital city is connected with each other city directly by the roads. Formally, if city $x$ is a capital city, then for every $1 \le i \le n,\ i \ne x$, there is a road between cities $x$ and $i$.
5. There is **at most one** road between any two cities.
6. Price of passing a road directly depends on beauty values of cities it connects. Thus if there is a road between cities $i$ and $j$, price of passing it equals $c_i \cdot c_j$.

Mishka started to gather her things for a trip, but didn't still decide which route to follow and thus she asked you to help her determine summary price of passing **each of the roads** in XXX. Formally, for every pair of cities $a$ and $b$ ($a < b$), such that there is a road between $a$ and $b$ you are to find sum of products $c_a \cdot c_b$. Will you help her?

## Input

The first line of the input contains two integers $n$ and $k$ ($3 \le n \le 100\,000$, $1 \le k \le n$) — the number of cities in XXX and the number of capital cities among them.

The second line of the input contains $n$ integers $c_1, c_2, \ldots, c_n$ ($1 \le c_i \le 10\,000$) — beauty values of the cities.

The third line of the input contains $k$ distinct integers $id_1, id_2, \ldots, id_k$ ($1 \le id_i \le n$) — indices of capital cities. Indices are given in ascending order.

## Output

Print the only integer — summary price of passing each of the roads in XXX.

**Examples**

| input |
| --- |
| 4 1<br>2 3 1 2<br>3 |

| output |
| --- |
| 17 |

| input |
| --- |
| 5 2<br>3 5 2 2 4<br>1 4 |

| output |
| --- |
| 71 |

## Note

This image describes first sample case:



It is easy to see that summary price is equal to $17$.

This image describes second sample case:



It is easy to see that summary price is equal to $71$.

# C. Chris and Road

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

And while Mishka is enjoying her trip...

Chris is a little brown bear. No one knows, where and when he met Mishka, but for a long time they are together (excluding her current trip). However, best friends are important too. John is Chris' best friend.

Once walking with his friend, John gave Chris the following problem:

At the infinite horizontal road of width $w$, bounded by lines $y = 0$ and $y = w$, there is a bus moving, presented as a convex polygon of $n$ vertices. The bus moves continuously with a constant speed of $v$ in a straight $Ox$ line in direction of decreasing $x$ coordinates, thus in time **only $x$ coordinates** of its points are changing. Formally, after time $t$ each of $x$ coordinates of its points will be decreased by $vt$.

There is a pedestrian in the point $(0, 0)$, who can move only by a vertical pedestrian crossing, presented as a segment connecting points $(0, 0)$ and $(0, w)$ with any speed not exceeding $u$. Thus the pedestrian can move only in a straight line $Oy$ in any direction with any speed not exceeding $u$ and not leaving the road borders. The pedestrian can instantly change his speed, thus, for example, he can stop instantly.

Please look at the sample note picture for better understanding.

We consider the pedestrian is *hit by the bus*, if at any moment the point he is located in lies **strictly inside** the bus polygon (this means that if the point lies on the polygon vertex or on its edge, the pedestrian is not hit by the bus).

You are given the bus position at the moment $0$. Please help Chris determine minimum amount of time the pedestrian needs to cross the road and reach the point $(0, w)$ and not to be hit by the bus.

## Input

The first line of the input contains four integers $n, w, v, u$ ($3 \le n \le 10\,000$, $1 \le w \le 10^9$, $1 \le v,\ u \le 1000$) — the number of the bus polygon vertices, road width, bus speed and pedestrian speed respectively.

The next $n$ lines describes polygon vertices in counter-clockwise order. $i$-th of them contains pair of integers $x_i$ and $y_i$ ($-10^9 \le x_i \le 10^9$, $0 \le y_i \le w$) — coordinates of $i$-th polygon point. It is guaranteed that the polygon is non-degenerate.

## Output

Print the single real $t$ — the time the pedestrian needs to croos the road and not to be hit by the bus. The answer is considered correct if its relative or absolute error doesn't exceed $10^{-6}$.

**Example**

| input |
| --- |
| 5 5 1 2<br>1 2<br>3 1<br>4 3<br>3 4<br>1 4 |
| output |
| 5.0000000000 |

## Note

Following image describes initial position in the first sample case:

# D. Mishka and Interesting sum

time limit per test: 3.5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little Mishka enjoys programming. Since her birthday has just passed, her friends decided to present her with array of non-negative integers $a_1, a_2, ..., a_n$ of $n$ elements!

Mishka loved the array and she instantly decided to determine its beauty value, but she is too little and can't process large arrays. Right because of that she invited you to visit her and asked you to process $m$ queries.

Each query is processed in the following way:

1. Two integers $l$ and $r$ ($1 \le l \le r \le n$) are specified — bounds of query segment.
2. Integers, presented in array segment $[l, r]$ (in sequence of integers $a_l, a_{l+1}, ..., a_r$) **even number of times**, are written down.
3. XOR-sum of written down integers is calculated, and this value is the answer for a query. Formally, if integers written down in point 2 are $x_1, x_2, ..., x_k$, then Mishka wants to know the value , where — operator of exclusive bitwise OR.

Since only the little bears know the definition of array beauty, all you are to do is to answer each of queries presented.

## Input

The first line of the input contains single integer $n$ ($1 \le n \le 1\,000\,000$) — the number of elements in the array.

The second line of the input contains $n$ integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^9$) — array elements.

The third line of the input contains single integer $m$ ($1 \le m \le 1\,000\,000$) — the number of queries.

Each of the next $m$ lines describes corresponding query by a pair of integers $l$ and $r$ ($1 \le l \le r \le n$) — the bounds of query segment.

## Output

Print $m$ non-negative integers — the answers for the queries in the order they appear in the input.

**Examples**

| input |
| --- |
| 3<br>3 7 8<br>1<br>1 3 |

| output |
| --- |
| 0 |

| input |
| --- |
| 7<br>1 2 1 3 3 2 3<br>5<br>4 7<br>4 5<br>1 3<br>1 7<br>1 5 |

| output |
| --- |
| 0<br>3<br>1<br>3<br>2 |

## Note

In the second sample:

There is no integers in the segment of the first query, presented even number of times in the segment — the answer is $0$.

In the second query there is only integer $3$ is presented even number of times — the answer is $3$.

In the third query only integer $1$ is written down — the answer is $1$.

In the fourth query all array elements are considered. Only $1$ and $2$ are presented there even number of times. The answer is .

In the fifth query $1$ and $3$ are written down. The answer is .

# E. Mishka and Divisors

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

After playing with her beautiful array, Mishka decided to learn some math. After learning how to multiply, divide and what is divisibility, she is now interested in solving the following problem.

You are given integer $k$ and array $a_1, a_2, ..., a_n$ of $n$ integers. You are to find **non-empty** subsequence of array elements such that the product of its elements is divisible by $k$ and it contains minimum possible number of elements.

Formally, you are to find a sequence of indices $1 \le i_1 < i_2 < ... < i_m \le n$ such that  is divisible by $k$ while $m$ is minimum possible among all such variants.

If there are more than one such subsequences, you should choose one among them, such that sum of its elements is **minimum possible**.

Mishka quickly solved this problem. Will you do so?

## Input

The first line of the input contains two integers $n$ and $k$ ($1 \le n \le 1\,000$, $1 \le k \le 10^{12}$).

The second line of the input contains $n$ integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^{12}$) — array elements.

## Output

Print single positive integer $m$ in the first line — the number of elements in desired sequence.

In the second line print $m$ distinct integers — the sequence of indices of given array elements, which should be taken into the desired sequence.

If there are more than one such subsequence (e.g. subsequence of minimum possible number of elements and with minimum possible sum of elements), you can print any of them.

If there are no such subsequences, print $-1$ in the only line.

**Example**

| input |
| --- |
| 5 60<br>2 4 6 5 2 |
| output |
| 3<br>4 3 1 |