A

Compass

time limit per test: 2 seconds
memory limit per test: 64 megabytes
input: standard input
output: standard output

The university of Algoland is located in a single huge building. It is great. It is the best building in the world. It only uses the best angles: the right angle! But the building is so huge that it is also very confusing to new students, because it is very easy to get lost.

The rector of Algoland's university, a former professor in physics, had a great idea to prevent students from getting lost in the future: He bought an incredibly strong magnet with the intention of placing it somewhere inside the building and using it as an emergency meeting point. On the first day of the semester, every student gets a free compass. With that compass the student can always tell the direction towards the magnet (the magnet is so strong that it completely dominates the earth's magnetic field). If a student gets lost, she can follow the following simple procedure to get to the emergency meeting point at the magnet's location:

- Move straight into the direction towards the magnet until you either reach the magnet, or bump into a wall.
- If your path is blocked by a wall, follow the direction alongside the wall that brings you closer towards the magnet, until you either reach the end of the wall or your path becomes orthogonal to the compass direction.
- If the wall is perfectly orthogonal to the compass direction, or you end up in a corner, you are stuck. Scream as loud as you can!

In the corner case where you want to walk parallel to a wall at the exact coordinate of a wall, you are not stopped by the wall. We assume here that you are infinitesimally small (which is true in proportion to the size of the building).

The rector now wants your help to place the magnet inside the building in such a way that every student can reach it (following the procedure above) no matter where inside the building the student gets lost. Actually, you just have to decide if this is possible or not.

Input

The first line contains an integer N, the number of points, $4 \leq N \leq 10^3$.

Each of the following N lines contains two coordinates x and y ($0 \leq x \leq 10^3$ and $0 \leq y \leq 10^3$). Each line describes a corner point of the single wall that delimits the inside from the outside of the building. The points are given in clockwise order and all the angles are guaranteed to be 90 degrees.

Output

Print a single line containing a single word: either `SAFETY` if it is possible to place the magnet such that one is always able to find it, or `DANGER` otherwise.

Examples

input
12 1 0 1 3 2 3 2 1 3 1 3 2 4 2 4 1 5 1 5 3 6 3

```
6 0
```

output

```
SAFETY
```

input

```
12
0 4
5 4
5 0
2 0
2 2
3 2
3 1
4 1
4 3
1 3
1 1
0 1
```

output

```
DANGER
```

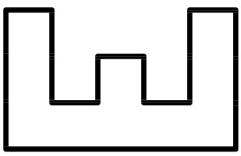
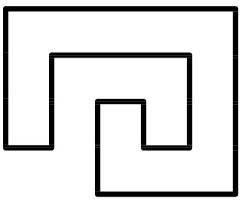
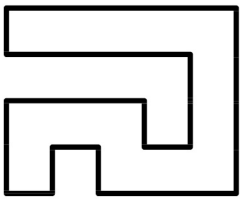
input

```
14
6 4
6 0
3 0
3 1
2 1
2 0
1 0
1 2
4 2
4 1
5 1
5 3
1 3
1 4
```

output

```
SAFETY
```

Note

B Police Hypothesis

time limit per test: 8 seconds
memory limit per test: 1024 megabytes
input: standard input
output: standard output

The public transport system of Nlogônia has an express network connecting the main points of interest of the country. There are $N - 1$ bullet trains connecting N attractions such that from one of the points of interest you can reach any other using only this network.

As anywhere in the world, it is common that there is graffiti on the train stations. What caught the attention of the police of the country is the fact that in each one of the stations it is possible to find exactly one letter pinned with a specific style. The hypothesis is that criminals may be changing the graffiti as a means of communication and, therefore, it was decided to create a system capable of monitoring the graffiti and its amendments.

Given a pattern P, the description of the connections between stations and the suspicious letters in each of them, your task is to write a program able to deal with the following operations:

- `1` u v: print how many times the pattern P occurs in the path from u to v if we look at the string of characters associated with the consecutive vertices of the path;
- `2` u x: change the suspicious letter at the station u to x.

Input

The first input line contains two integers N and Q ($1 \leq N, Q \leq 10^5$), representing the number of stations and the number of transactions that must be processed. The second line contains the pattern P monitored ($1 \leq |P| \leq 100$). The third line contains a string with N characters representing the letters initially associated with each of the stations. Each of the following $N - 1$ lines contains two integers u and v indicating that there is a bullet train between u and v. The following Q lines describe the operations that must be processed as described above.

Output

Your program should print a line for each operation of type 1 containing an integer that represents the number of occurrences of the pattern P on the way.

Examples

input
4 4 xtc xtzy 1 2 2 3 3 4 1 1 3 2 3 c 1 1 3 1 3 1

output
0 1 0

input
6 7 lol dlorlx 1 2 1 3

```
3 4
3 5
5 6
1 2 6
2 3 l
2 6 l
2 5 o
1 2 6
2 1 o
1 6 2
```

output

```
0
1
2
```

input

```
5 2
aba
ababa
1 2
2 3
3 4
4 5
1 1 5
1 5 1
```

output

```
2
2
```

C Conveyor

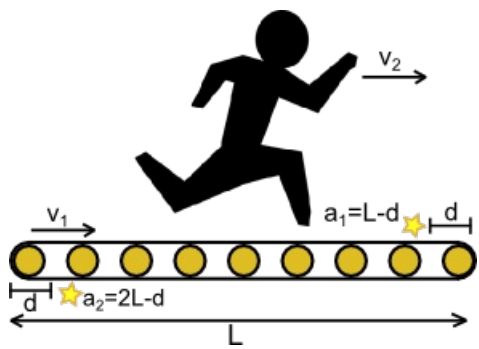
time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Anton came to a chocolate factory. There he found a working conveyor and decided to run on it from the beginning to the end.

The conveyor is a looped belt with a total length of $2l$ meters, of which l meters are located on the surface and are arranged in a straight line. The part of the belt which turns at any moment (the part which emerges from under the floor to the surface and returns from the surface under the floor) is assumed to be negligibly short.

The belt is moving uniformly at speed v_1 meters per second. Anton will be moving on it in the same direction at the constant speed of v_2 meters per second, so his speed relatively to the floor will be $v_1 + v_2$ meters per second. Anton will neither stop nor change the speed or the direction of movement.

Here and there there are chocolates stuck to the belt (n chocolates). They move together with the belt, and do not come off it. Anton is keen on the chocolates, but he is more keen to move forward. So he will pick up all the chocolates he will pass by, but nothing more. If a chocolate is at the beginning of the belt at the moment when Anton starts running, he will take it, and if a chocolate is at the end of the belt at the moment when Anton comes off the belt, he will leave it.



The figure shows an example with two chocolates. One is located in the position $a_1 = l - d$, and is now on the top half of the belt, the second one is in the position $a_2 = 2l - d$, and is now on the bottom half of the belt.

You are given the positions of the chocolates relative to the initial start position of the belt $0 \le a_1 < a_2 < ... < a_n < 2l$. The positions on the belt from 0 to l correspond to the top, and from l to $2l$ — to the the bottom half of the belt (see example). All coordinates are given in meters.

Anton begins to run along the belt at a random moment of time. This means that all possible positions of the belt at the moment he starts running are equiprobable. For each i from 0 to n calculate the probability that Anton will pick up exactly i chocolates.

Input

The first line contains space-separated integers n, l, v_1 and v_2 ($1 \le n \le 10^5$, $1 \le l, v_1, v_2 \le 10^9$) — the number of the chocolates, the length of the conveyor's visible part, the conveyor's speed and Anton's speed.

The second line contains a sequence of space-separated integers $a_1, a_2, ..., a_n$ ($0 \le a_1 < a_2 < ... < a_n < 2l$) — the coordinates of the chocolates.

Output

Print $n + 1$ numbers (one per line): the probabilities that Anton picks up exactly i chocolates, for each i from 0 (the first line) to n (the last line). The answer will be considered correct if each

number will have absolute or relative error of at most than 10^{-9}.

Examples

input
1 1 1 1 0
output
0.7500000000000000000 0.2500000000000000000

input
2 3 1 2 2 5
output
0.3333333333333331000 0.6666666666666663000 0.0000000000000000000

Note

In the first sample test Anton can pick up a chocolate if by the moment he starts running its coordinate is less than 0.5; but if by the moment the boy starts running the chocolate's coordinate is greater than or equal to 0.5, then Anton won't be able to pick it up. As all positions of the belt are equiprobable, the probability of picking up the chocolate equals $\frac{0.5}{2} = 0.25$, and the probability of not picking it up equals $\frac{1.5}{2} = 0.75$.

D

Bug in Code

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

Recently a serious bug has been found in the FOS code. The head of the F company wants to find the culprit and punish him. For that, he set up an organizational meeting, the issue is: who's bugged the code? Each of the n coders on the meeting said: 'I know for sure that either x or y did it!'

The head of the company decided to choose two suspects and invite them to his office. Naturally, he should consider the coders' opinions. That's why the head wants to make such a choice that at least p of n coders agreed with it. A coder agrees with the choice of two suspects if at least one of the two people that he named at the meeting was chosen as a suspect. In how many ways can the head of F choose two suspects?

Note that even if some coder was chosen as a suspect, he can agree with the head's choice if he named the other chosen coder at the meeting.

Input

The first line contains integers n and p $(3 \le n \le 3 \cdot 10^5; 0 \le p \le n)$ — the number of coders in the F company and the minimum number of agreed people.

Each of the next n lines contains two integers x_i, y_i $(1 \le x_i, y_i \le n)$ — the numbers of coders named by the i-th coder. It is guaranteed that $x_i \ne i,\ y_i \ne i,\ x_i \ne y_i$.

Output

Print a single integer — the number of possible two-suspect sets. Note that the order of the suspects doesn't matter, that is, sets $(1, 2)$ и $(2, 1)$ are considered identical.

Examples

input
4 2 2 3 1 4 1 4 2 1

output
6

input
8 6 5 6 5 7 5 8 6 2 2 1 7 3 1 3 1 4

output
1

The only programming contests Web 2.0 platform
Server time: Jul/16/2020 15:21:44 (i3).
Desktop version, switch to mobile version.
Privacy Policy

Supported by

E The hat

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

This is an interactive problem.

Imur Ishakov decided to organize a club for people who love to play the famous game «The hat». The club was visited by n students, where n is even. Imur arranged them all in a circle and held a draw to break the students in pairs, but something went wrong. The participants are numbered so that participant i and participant $i + 1$ ($1 \le i \le n$ - 1) are adjacent, as well as participant n and participant 1. Each student was given a piece of paper with a number in such a way, that for every two adjacent students, these numbers differ exactly by one. The plan was to form students with the same numbers in a pair, but it turned out that not all numbers appeared exactly twice.

As you know, the most convenient is to explain the words to the partner when he is sitting exactly across you. Students with numbers i and $i + \frac{n}{2}$ sit across each other. Imur is wondering if there are two people sitting across each other with the same numbers given. Help him to find such pair of people if it exists.

You can ask questions of form «which number was received by student i?», and the goal is to determine whether the desired pair exists in no more than 60 questions.

Input

At the beginning the even integer n ($2 \le n \le 100\,000$) is given — the total number of students.

You are allowed to ask no more than 60 questions.

Output

To ask the question about the student i ($1 \le i \le n$), you should print «`? i`». Then from standard output you can read the number a_i received by student i (- $10^9 \le a_i \le 10^9$).

When you find the desired pair, you should print «`! i`», where i is any student who belongs to the pair ($1 \le i \le n$). If you determined that such pair doesn't exist, you should output «`! -1`». In both cases you should immediately terminate the program.

The query that contains your answer is not counted towards the limit of 60 queries.

Please make sure to flush the standard output after each command. For example, in C++ use function `fflush(stdout)`, in Java call `System.out.flush()`, in Pascal use `flush(output)` and `stdout.flush()` for Python language.

Hacking

Use the following format for hacking:

In the first line, print one even integer n ($2 \le n \le 100\,000$) — the total number of students.

In the second line print n integers a_i (- $10^9 \le a_i \le 10^9$) separated by spaces, where a_i is the number to give to i-th student. Any two adjacent elements, including n and 1, must differ by 1 or - 1.

The hacked solution will not have direct access to the sequence a_i.

Examples

input
8

```
2
2
```

output

```
? 4
? 8
! 4
```

input

```
6
1
2
3
2
1
0
```

output

```
? 1
? 2
? 3
? 4
? 5
? 6
! -1
```

Note

Input-output in statements illustrates example interaction.

In the first sample the selected sequence is $1, 2, 1, 2, 3, 4, 3, 2$

In the second sample the selection sequence is $1, 2, 3, 2, 1, 0$.

F Looking for Order

time limit per test: 4 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Girl Lena likes it when everything is in order, and looks for order everywhere. Once she was getting ready for the University and noticed that the room was in a mess — all the objects from her handbag were thrown about the room. Of course, she wanted to put them back into her handbag. The problem is that the girl cannot carry more than two objects at a time, and cannot move the handbag. Also, if he has taken an object, she cannot put it anywhere except her handbag — her inherent sense of order does not let her do so.

You are given the coordinates of the handbag and the coordinates of the objects in some Cartesian coordinate system. It is known that the girl covers the distance between any two objects in the time equal to the squared length of the segment between the points of the objects. It is also known that initially the coordinates of the girl and the handbag are the same. You are asked to find such an order of actions, that the girl can put all the objects back into her handbag in a minimum time period.

Input

The first line of the input file contains the handbag's coordinates x_s, y_s. The second line contains number n ($1 \le n \le 24$) — the amount of objects the girl has. The following n lines contain the objects' coordinates. All the coordinates do not exceed 100 in absolute value. All the given positions are different. All the numbers are integer.

Output

In the first line output the only number — the minimum time the girl needs to put the objects into her handbag.

In the second line output the possible optimum way for Lena. Each object in the input is described by its index number (from 1 to n), the handbag's point is described by number 0. The path should start and end in the handbag's point. If there are several optimal paths, print any of them.

Examples

input
0 0 2 1 1 -1 1
output
8 0 1 2 0

input
1 1 3 4 3 3 4 0 0
output
32 0 1 2 0 3 0

The only programming contests Web 2.0 platform
Server time: Jul/16/2020 15:17:57 (i3).
Desktop version, switch to mobile version.
Privacy Policy

Supported by

G Leaving the bar

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

For a vector $v = (x, y)$, define $|v| = \sqrt{x^2 + y^2}$.

Allen had a bit too much to drink at the bar, which is at the origin. There are n vectors $\overrightarrow{v_1}, \overrightarrow{v_2}, \cdots, \overrightarrow{v_n}$. Allen will make n moves. As Allen's sense of direction is impaired, during the i-th move he will either move in the direction $\overrightarrow{v_i}$ or $-\overrightarrow{v_i}$. In other words, if his position is currently $p = (x, y)$, he will either move to $p + \overrightarrow{v_i}$ or $p - \overrightarrow{v_i}$.

Allen doesn't want to wander too far from home (which happens to also be the bar). You need to help him figure out a sequence of moves (a sequence of signs for the vectors) such that his final position p satisfies $|p| \leq 1.5 \cdot 10^6$ so that he can stay safe.

Input

The first line contains a single integer n ($1 \leq n \leq 10^5$) — the number of moves.

Each of the following lines contains two space-separated integers x_i and y_i, meaning that $\overrightarrow{v_i} = (x_i, y_i)$. We have that $|v_i| \leq 10^6$ for all i.

Output

Output a single line containing n integers $c_1, c_2, \cdots, c_n$, each of which is either 1 or -1. Your solution is correct if the value of $p = \sum_{i=1}^{n} c_i \overrightarrow{v_i}$, satisfies $|p| \leq 1.5 \cdot 10^6$.

It can be shown that a solution always exists under the given constraints.

Examples

input
3 999999 0 0 999999 999999 0
output
1 1 -1

input
1 -824590 246031
output
1

input
8 -67761 603277 640586 -396671 46147 -122580 569609 -2112 400 914208 131792 309779 -850150 -486293 5272 721899
output

```
1 1 1 1 1 1 1 -1
```

H Salazar Slytherin's Locket

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

Harry came to know from Dumbledore that Salazar Slytherin's locket is a horcrux. This locket was present earlier at 12 Grimmauld Place, the home of Sirius Black's mother. It was stolen from there and is now present in the Ministry of Magic in the office of Dolorous Umbridge, Harry's former Defense Against the Dark Arts teacher.

Harry, Ron and Hermione are infiltrating the Ministry. Upon reaching Umbridge's office, they observed a code lock with a puzzle asking them to calculate count of magic numbers between two integers l and r (both inclusive).

Harry remembered from his detention time with Umbridge that she defined a magic number as a number which when converted to a given base b, all the digits from 0 to b - 1 appear even number of times in its representation without any leading zeros.

You have to answer q queries to unlock the office. Each query has three integers b_i, l_i and r_i, the base and the range for which you have to find the count of magic numbers.

Input

First line of input contains q ($1 \le q \le 10^5$) — number of queries.

Each of the next q lines contain three space separated integers b_i, l_i, r_i ($2 \le b_i \le 10$, $1 \le l_i \le r_i \le 10^{18}$).

Output

You have to output q lines, each containing a single integer, the answer to the corresponding query.

Examples

input
2 2 4 9 3 1 10
output
1 2

input
2 2 1 100 5 1 100
output
21 4

Note

In sample test case 1, for first query, when we convert numbers 4 to 9 into base 2, we get:

- $4 = 100_2$,
- $5 = 101_2$,
- $6 = 110_2$,
- $7 = 111_2$,

- $8 = 1000_2$,
- $9 = 1001_2$.

Out of these, only base 2 representation of 9 has even number of 1 and 0. Thus, the answer is 1.

I Multicolored Marbles

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

Polycarpus plays with red and blue marbles. He put n marbles from the left to the right in a row. As it turned out, the marbles form a *zebroid*.

A non-empty sequence of red and blue marbles is a *zebroid*, if the colors of the marbles in this sequence alternate. For example, sequences (`red; blue; red`) and (`blue`) are zebroids and sequence (`red; red`) is not a zebroid.

Now Polycarpus wonders, how many ways there are to pick a zebroid **subsequence** from this sequence. Help him solve the problem, find the number of ways modulo 1000000007 $(10^9 + 7)$.

Input

The first line contains a single integer n $(1 \le n \le 10^6)$ — the number of marbles in Polycarpus's sequence.

Output

Print a single number — the answer to the problem modulo 1000000007 $(10^9 + 7)$.

Examples

input
3
output
6

input
4
output
11

Note

Let's consider the first test sample. Let's assume that Polycarpus initially had sequence (`red; blue; red`), so there are six ways to pick a zebroid:

- pick the first marble;
- pick the second marble;
- pick the third marble;
- pick the first and second marbles;
- pick the second and third marbles;
- pick the first, second and third marbles.

It can be proven that if Polycarpus picks (`blue; red; blue`) as the initial sequence, the number of ways won't change.

Privacy Policy

Supported by

J

Numbers

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

Furik loves writing all sorts of problems, especially such that he can't solve himself. You've got one of his problems, the one Furik gave to Rubik. And Rubik asks you to solve it.

There is integer n and array a, consisting of ten integers, indexed by numbers from 0 to 9. Your task is to count the number of positive integers with the following properties:

- the number's length does not exceed n;
- the number doesn't have leading zeroes;
- digit i $(0 \le i \le 9)$ occurs in the number at least $a[i]$ times.

Input

The first line contains integer n $(1 \le n \le 100)$. The next line contains 10 integers $a[0]$, $a[1]$, ..., $a[9]$ $(0 \le a[i] \le 100)$ — elements of array a. The numbers are separated by spaces.

Output

On a single line print the remainder of dividing the answer to the problem by 1000000007 $(10^9 + 7)$.

Examples

input
1 0 0 0 0 0 0 0 0 0 1
output
1

input
2 1 1 0 0 0 0 0 0 0 0
output
1

input
3 1 1 0 0 0 0 0 0 0 0
output
36

Note

In the first sample number 9 meets the requirements.

In the second sample number 10 meets the requirements.

In the third sample numbers **10, 110, 210, 120, 103** meet the requirements. There are other suitable numbers, 36 in total.

The only programming contests Web 2.0 platform
Server time: Jul/16/2020 15:21:26 (i3).
Desktop version, switch to mobile version.
Privacy Policy

Supported by

K The Fair Nut and Rectangles

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

The Fair Nut got stacked in planar world. He should solve this task to get out.

You are given n rectangles with vertexes in $(0, 0)$, $(x_i, 0)$, (x_i, y_i), $(0, y_i)$. For each rectangle, you are also given a number a_i. Choose some of them that the area of union minus sum of a_i of the chosen ones is maximum.

It is guaranteed that there are no nested rectangles.

Nut has no idea how to find the answer, so he asked for your help.

Input

The first line contains one integer n ($1 \le n \le 10^6$) — the number of rectangles.

Each of the next n lines contains three integers x_i, y_i and a_i ($1 \le x_i, y_i \le 10^9$, $0 \le a_i \le x_i \cdot y_i$).

It is guaranteed that there are no nested rectangles.

Output

In a single line print the answer to the problem — the maximum value which you can achieve.

Examples

input
3 4 4 8 1 5 0 5 2 10
output
9

input
4 6 2 4 1 6 2 2 4 3 5 3 8
output
10

Note

In the first example, the right answer can be achieved by choosing the first and the second rectangles.

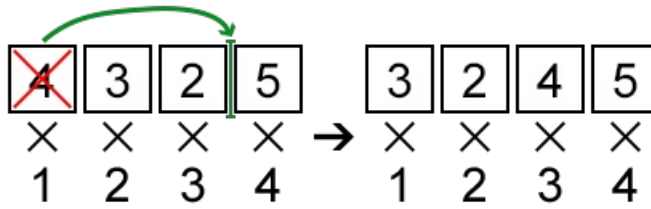
In the second example, the right answer can also be achieved by choosing the first and the second rectangles.

L Product Sum

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

Blake is the boss of Kris, however, this doesn't spoil their friendship. They often gather at the bar to talk about intriguing problems about maximising some values. This time the problem is really special.

You are given an array a of length n. The *characteristic* of this array is the value $c = \sum_{i=1}^{n} a_i \cdot i$ — the sum of the products of the values a_i by i. One may perform the following operation **exactly once**: pick some element of the array and move to any position. In particular, it's allowed to move the element to the beginning or to the end of the array. Also, it's allowed to put it back to the initial position. The goal is to get the array with the maximum possible value of characteristic.



Input

The first line of the input contains a single integer n ($2 \le n \le 200\,000$) — the size of the array a.

The second line contains n integers a_i ($1 \le i \le n$, $|a_i| \le 1\,000\,000$) — the elements of the array a.

Output

Print a single integer — the maximum possible value of characteristic of a that can be obtained by performing no more than one move.

Examples

input
4 4 3 2 5
output
39

input
5 1 1 2 7 1
output
49

input
3 1 1 2
output
9

Note

In the first sample, one may pick the first element and place it before the third (before 5). Thus, the answer will be $3 \cdot 1 + 2 \cdot 2 + 4 \cdot 3 + 5 \cdot 4 = 39$.

In the second sample, one may pick the fifth element of the array and place it before the third. The answer will be $1 \cdot 1 + 1 \cdot 2 + 1 \cdot 3 + 2 \cdot 4 + 7 \cdot 5 = 49$.
