

Problem A. Spaceship

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 256 megabytes

Space Ranger is caught at an alien spaceship. He is surrounded by enemies. To break free he needs to destroy all enemies in some particular order.

There are n enemies around Space Ranger, the i -th of them has power f_i . To break free Space Ranger needs to destroy all enemies in such order, that the power of the last destroyed enemy is equal to the sum of powers of all other enemies.

Space Ranger is short of time, so he has failed to find the required order. Help him! Find the way to destroy all enemies and break free.

Input

The first line of input contains n — the number of enemies ($2 \leq n \leq 10^5$). The second line contains n integers f_i — powers of the enemies ($-10^9 \leq f_i \leq 10^9$).

Output

Output values f_i in order in which the enemies must be destroyed. If there are several ways to destroy all enemies and break free, print any one. It is guaranteed that there is at least one way to do it.

Examples

standard input	standard output
3 2 5 3	2 3 5
5 -1 1 0 1 -1	-1 1 1 -1 0

Problem B. Rangers in the Bus

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

Space Rangers are going to a secret meeting by a bus.

The Main Villain is watching the bus that he thinks the rangers are using. The bus has n rows of seats, each row has two seats: one to the left of the passage, one to the right. Rows are numbered from 1 to n , starting from the front of the bus. There were k passengers that entered the bus at the first stop, the villain knows the order they entered, and who took which seat. Initially all seats were free.

He knows the way rangers choose the seat when they enter the bus:

- Red Ranger likes front seats. When he enters the bus he always chooses the free seat in the minimal row. If both seats are free, he chooses the left one.
- Blue Ranger also likes front seats. But unlike the Red Ranger, he chooses the right seat if both seats in the minimal possible row are free.
- Black Ranger likes back seats. He always chooses the free seat in the maximal row. If both seats in that row are free, he chooses the left one.
- Yellow Ranger also likes back seats, but chooses the right one if possible.
- Pink Ranger has no preferences and just chooses any seat.

For each ranger the villain wants to know who of the k passengers could be that ranger. Note that some rangers could take another bus.

Input

The first line of input contains two integers: n and k — the number of rows and the number of passengers ($1 \leq n \leq 10^9, 1 \leq k \leq \min(2 \cdot 10^5, 2 \cdot n)$).

The following k lines describe passengers in order they entered the bus.

The i -th of these lines contains integers x_i and y_i — the row that the i -th passenger chose and the seat in that row ($1 \leq x_i \leq n, 1 \leq y_i \leq 2$), x_i is the row number, $y_i = 1$ means that the passenger took the left seat, $y_i = 2$ — the right seat.

Seats for different passengers are distinct.

Output

The first line must contain s_1 — the number of passengers that could be Red Ranger, followed by a space and s_1 space separated integers, the numbers of there passengers in order they entered the bus (the passengers are numbered from 1 to k in order they entered the bus).

The following four lines must contain the same information about Blue Ranger, Black Ranger, Yellow Ranger, and Pink Ranger, correspondingly.

Example

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

Problem C. Magic Weapon

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

To create the magic weapon Space Rangers need to use three details: green, red and blue.

Green, Red and Blue rangers want to know how many different ways they have to create the weapon. Each ranger has several details, Green Ranger has green details, Red Ranger has red details, Blue Ranger has blue details.

Each detail has its model number. To create the weapon rangers need to follow three rules:

- The first digit of the model number of the red detail must be equal to the last digit of the model number of the green detail.
- The last digit of the model number of the red detail must be equal to the first digit of the model number of the blue detail.
- Model numbers of all three details must be different.

Two ways to create the weapon are considered different if at least one ranger uses another detail, even if its model number is the same.

For each ranger you are given model numbers of their details. One ranger can have several details with the same model number.

Help rangers to find out how many different ways to create the magic weapon they have.

Input

The first line of input contains three integers: g , r and b — the number of details that Green Ranger, Red Ranger and Blue Ranger have, correspondingly ($1 \leq g, r, b \leq 10^5$).

The following line contains g integers x_i — model numbers of the green details ($0 \leq x_i \leq 10^9$).

The following line contains r integers y_i — model numbers of the red details ($0 \leq y_i \leq 10^9$).

The following line contains b integers z_i — model numbers of the blue details ($0 \leq z_i \leq 10^9$).

Output

Print one integer — the number of ways to create the magic weapon.

Example

standard input	standard output
3 3 2 101 11 52 11 23 23 31 13	3

Problem D. Knights and Knaves

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

Space Ranger has arranged his army in two rows of k soldiers. Neighbors of a soldier in this arrangement are soldiers to the left of it in the same row, to the right of it in the same row, and at the same position in the opposite row.

Ranger knows that there are knights that always tell truth, and knaves that lie to at least one question asked in his army.

Ranger chose one or two questions from the set:

- “Is it true that there are exactly x knights among your neighbors?”
- “Is it true that there are exactly y knaves among your neighbors?”

and asked each soldier. Each soldier was asked questions with the same values of x and/or y .

Suddenly all soldiers answers “yes” to all questions asked.

Now Space Ranger wants to know what minimal and maximal number of knights can be there in his army. Help him to find out.

Let us consider the second sample test. Each row has 5 soldiers, the questions asked are “Is it true that exactly one of your neighbors is a knight?” and “Is it true that exactly two of your neighbors are knaves?”

First, all soldiers can be knaves (they lied to the first question in this case, though the first and the last soldier in each row told the truth to the second question, but at least one lie is required). This variant gives the minimal number of knights: 0. Another variant is that there are two knights in each row: at the 2-nd and the 4-th position. In this case they told the truth to both questions, and others lied answering to the second question (each knave now has one knave neighbor). This variant gives the maximal number of knights: 4.

Input

The input consists of one row that contains three integers: k, x, y — number of soldiers in each row, and the parameters of the questions ($1 \leq k \leq 10^5$, $-1 \leq x, y \leq 3$).

If $x = -1$ Space Ranger didn't ask the first question.

If $y = -1$ Space Ranger didn't ask the second question.

It is guaranteed that at least one question was asked.

Output

If there is no possible answer for the given k, x and y , output -1.

In the other case two integers — the minimal possible number of knights in the army and the maximal possible number of knights in the army.

Examples

standard input	standard output
2 0 -1	2 2
5 1 2	0 4
1 2 2	0 0
10 0 3	5 8

Problem E. Parallelepiped

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

Black Ranger wants to create a rectangular parallelepiped. To do it he plans to use 6 of n rectangular metal sheets he has. The i -th sheet has size a_i times b_i meters.

Each face of the parallelepiped must be a solid metal sheet. The sheets must not be bent or cut and must not stand out of its limits. The sheets can be rotated if necessary.

Black Ranger wants to create a parallelepiped with the maximal possible volume. Help him.

Input

The first line contains one integer n — the number of metal sheets Black Ranger has ($6 \leq n \leq 2 \cdot 10^5$).

The following n lines contain two pairs of integers each: a_i, b_i — size of the i -th metal sheet ($1 \leq a_i, b_i \leq 10^6$).

Output

Output one integer — the maximal possible volume of a parallelepiped that can be created using the given metal sheets. If no parallelepiped can be created, output -1.

Examples

standard input	standard output
6 3 6 6 9 9 3 6 3 3 9 9 6	162
6 1 1 1 1 1 1 1 1 1 1 1 1	1
6 1 2 2 3 3 4 4 5 5 6 6 1	-1