A. Hit the Lottery

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

Allen has a LOT of money. He has n dollars in the bank. For security reasons, he wants to withdraw it in cash (we will not disclose the reasons here). The denominations for dollar bills are 1, 5, 10, 20, 100. What is the minimum number of bills Allen could receive after withdrawing his entire balance?

Input

The first and only line of input contains a single integer n ($1 \le n \le 10^9$).

Output

Output the minimum number of bills that Allen could receive.

Examples

input
```
125
```
output
```
3
```

input
```
43
```
output
```
5
```

input
```
1000000000
```
output
```
10000000
```

Note

In the first sample case, Allen can withdraw this with a 100 dollar bill, a 20 dollar bill, and a 5 dollar bill. There is no way for Allen to receive 125 dollars in one or two bills.

In the second sample case, Allen can withdraw two 20 dollar bills and three 1 dollar bills.

In the third sample case, Allen can withdraw 100000000 (ten million!) 100 dollar bills.

B. World Cup

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

Allen wants to enter a fan zone that occupies a round square and has n entrances.

There already is a queue of a_i people in front of the i-th entrance. Each entrance allows one person from its queue to enter the fan zone in one minute.

Allen uses the following strategy to enter the fan zone:

- Initially he stands in the end of the queue in front of the first entrance.
- Each minute, if he is not allowed into the fan zone during the minute (meaning he is not the first in the queue), he leaves the current queue and stands in the end of the queue of the next entrance (or the first entrance if he leaves the last entrance).

Determine the entrance through which Allen will finally enter the fan zone.

Input

The first line contains a single integer n ($2 \le n \le 10^5$) — the number of entrances.

The second line contains $$$n$$$ integers $$$a_1, a_2, \ldots, a_n$$$ ($$$0 \le a_i \le 10^9$$$) — the number of people in queues. These numbers do not include Allen.

Output

Print a single integer — the number of entrance that Allen will use.

Examples

input

```
4
2 3 2 0
```

output

```
3
```

input

```
2
10 10
```

output

```
1
```

input

```
6
5 2 6 5 7 4
```

output

```
6
```

Note

In the first example the number of people (not including Allen) changes as follows: $$$[\textbf{2}, 3, 2, 0] \to [1, \textbf{2}, 1, 0] \to [0, 1, \textbf{0}, 0]$$$. The number in bold is the queue Alles stands in. We see that he will enter the fan zone through the third entrance.

In the second example the number of people (not including Allen) changes as follows: $$$[\textbf{10}, 10] \to [9, \textbf{9}] \to [\textbf{8}, 8] \to [7, \textbf{7}] \to [\textbf{6}, 6] \to \\ [5, \textbf{5}] \to [\textbf{4}, 4] \to [3, \textbf{3}] \to [\textbf{2}, 2] \to [1, \textbf{1}] \to [\textbf{0}, 0]$$$.

In the third example the number of people (not including Allen) changes as follows: $$$[\textbf{5}, 2, 6, 5, 7, 4] \to [4, \textbf{1}, 5, 4, 6, 3] \to [3, 0, \textbf{4}, 3, 5, 2] \to \\ [2, 0, 3, \textbf{2}, 4, 1] \to [1, 0, 2, 1, \textbf{3}, 0] \to [0, 0, 1, 0, 2, \textbf{0}]$$$.

C. Tesla

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Allen dreams of one day owning a enormous fleet of electric cars, the car of the future! He knows that this will give him a big status boost. As Allen is planning out all of the different types of cars he will own and how he will arrange them, he realizes that he has a problem.

Allen's future parking lot can be represented as a rectangle with $$$4$$$ rows and $$$n$$$ ($$$n \le 50$$$) columns of rectangular spaces, each of which can contain at most one car at any time. He imagines having $$$k$$$ ($$$k \le 2n$$$) cars in the grid, and all the cars are initially in the second and third rows. Each of the cars also has a different designated parking space in the first or fourth row. Allen has to put the cars into corresponding parking places.

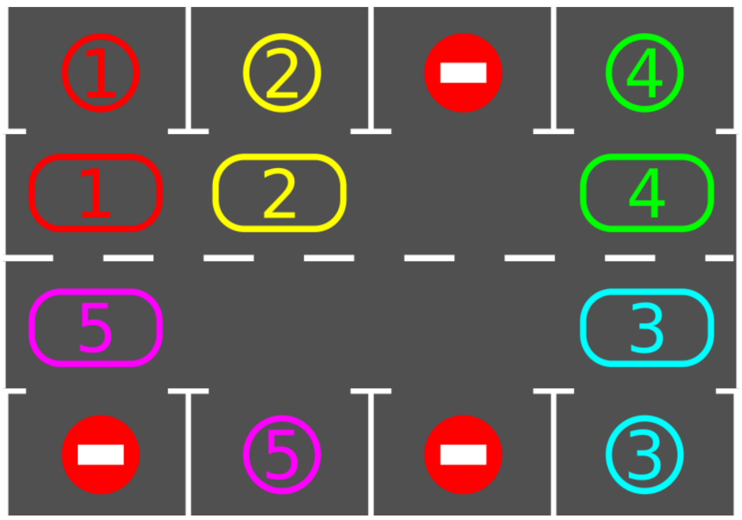

Illustration to the first example.

However, since Allen would never entrust his cars to anyone else, only one car can be moved at a time. He can drive a car from a space in any of the four cardinal directions to a neighboring empty space. Furthermore, Allen can only move one of his cars into a space on the first or fourth rows if it is the car's designated parking space.

Allen knows he will be a very busy man, and will only have time to move cars at most $$$20000$$$ times before he realizes that moving cars is not worth his time. Help Allen determine if he should bother parking his cars or leave it to someone less important.

Input

The first line of the input contains two space-separated integers $$$n$$$ and $$$k$$$ ($$$1 \le n \le 50$$$, $$$1 \le k \le 2n$$$), representing the number of columns and the number of cars, respectively.

The next four lines will contain $$$n$$$ integers each between $$$0$$$ and $$$k$$$ inclusive, representing the initial state of the parking lot. The rows are numbered $$$1$$$ to $$$4$$$ from top to bottom and the columns are numbered $$$1$$$ to $$$n$$$ from left to right.

In the first and last line, an integer $$$1 \le x \le k$$$ represents a parking spot assigned to car $$$x$$$ (you can only move this car to this place), while the integer $$$0$$$ represents a empty space (you can't move any car to this place).

In the second and third line, an integer $$$1 \le x \le k$$$ represents initial position of car $$$x$$$, while the integer $$$0$$$ represents an empty space (you can move any car to this place).

Each $$$x$$$ between $$$1$$$ and $$$k$$$ appears exactly once in the second and third line, and exactly once in the first and fourth line.

Output

If there is a sequence of moves that brings all of the cars to their parking spaces, with at most $$$20000$$$ car moves, then print $$$m$$$, the number of moves, on the first line. On the following $$$m$$$ lines, print the moves (one move per line) in the format $$$i$$$ $$$r$$$ $$$c$$$, which corresponds to Allen moving car $$$i$$$ to the neighboring space at row $$$r$$$ and column $$$c$$$.

If it is not possible for Allen to move all the cars to the correct spaces with at most $$$20000$$$ car moves, print a single line with the integer $$$-1$$$.

Examples

input

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

output

```
6
1 1 1
2 1 2
4 1 4
3 4 4
5 3 2
5 4 2
```

input

```
1 2
1
2
1
2
```

output

```
-1
```

input

```
1 2
1
1
2
2
```

output

```
2
1 1 1
2 4 1
```

Note

In the first sample test case, all cars are in front of their spots except car $$$5$$$, which is in front of the parking spot adjacent. The example shows the shortest possible sequence of moves, but any sequence of length at most $$$20000$$$ will be accepted.

In the second sample test case, there is only one column, and the cars are in the wrong order, so no cars can move and the task is impossible.

D. Suit and Tie

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Allen is hosting a formal dinner party. $2n$ people come to the event in n pairs (couples). After a night of fun, Allen wants to line everyone up for a final picture. The $2n$ people line up, but Allen doesn't like the ordering. Allen prefers if each pair occupies adjacent positions in the line, as this makes the picture more aesthetic.

Help Allen find the minimum number of swaps of **adjacent** positions he must perform to make it so that each couple occupies adjacent positions in the line.

Input

The first line contains a single integer n ($1 \le n \le 100$), the number of pairs of people.

The second line contains $2n$ integers $a_1, a_2, \dots, a_{2n}$. For each i with $1 \le i \le n$, i appears exactly twice. If $a_j = a_k = i$, that means that the j-th and k-th people in the line form a couple.

Output

Output a single integer, representing the minimum number of adjacent swaps needed to line the people up so that each pair occupies adjacent positions.

Examples

input
```
4
1 1 2 3 3 2 4 4
```
output
```
2
```

input
```
3
1 1 2 2 3 3
```
output
```
0
```

input
```
3
3 1 2 3 1 2
```
output
```
3
```

Note

In the first sample case, we can transform $1 1 2 3 3 2 4 4 \rightarrow 1 1 2 3 2 3 4 4 \rightarrow 1 1 2 2 3 3 4 4$ in two steps. Note that the sequence $1 1 2 3 3 2 4 4 \rightarrow 1 1 3 2 3 2 4 4 \rightarrow 1 1 3 3 2 2 4 4$ also works in the same number of steps.

The second sample case already satisfies the constraints; therefore we need 0 swaps.

E. Leaving the Bar

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

For a vector $\vec{v} = (x, y)$, 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 $\vec{v_1}, \vec{v_2}, \cdots, \vec{v_n}$. 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 $\vec{v_i}$ or $-\vec{v_i}$. In other words, if his position is currently $p = (x, y)$, he will either move to $p + \vec{v_i}$ or $p - \vec{v_i}$.

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 $|p| \le 1.5 \cdot 10^6$ so that he can stay safe.

Input

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

Each of the following lines contains two space-separated integers x_i and y_i, meaning that $\vec{v_i} = (x_i, y_i)$. We have that $|v_i| \le 10^6$ 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 $p = \sum_{i = 1}^n c_i \vec{v_i}$, satisfies $|p| \le 1.5 \cdot 10^6$.

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
```

F. Game

time limit per test: 3 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Allen and Bessie are playing a simple number game. They both know a function $f: \{0, 1\}^n \to \mathbb{R}$, i. e. the function takes n binary arguments and returns a real value. At the start of the game, the variables $x_1, x_2, \dots, x_n$ are all set to -1. Each round, with equal probability, one of Allen or Bessie gets to make a move. A move consists of picking an i such that $x_i = -1$ and either setting $x_i \to 0$ or $x_i \to 1$.

After n rounds all variables are set, and the game value resolves to $f(x_1, x_2, \dots, x_n)$. Allen wants to maximize the game value, and Bessie wants to minimize it.

Your goal is to help Allen and Bessie find the expected game value! They will play $r+1$ times though, so between each game, exactly one value of f changes. In other words, between rounds i and $i+1$ for $1 \le i \le r$, $f(z_1, \dots, z_n) \to g_i$ for some $(z_1, \dots, z_n) \in \{0, 1\}^n$. You are to find the expected game value in the beginning and after each change.

Input

The first line contains two integers n and r ($1 \le n \le 18$, $0 \le r \le 2^{18}$).

The next line contains 2^n integers $c_0, c_1, \dots, c_{2^n-1}$ ($0 \le c_i \le 10^9$), denoting the initial values of f. More specifically, $f(x_0, x_1, \dots, x_{n-1}) = c_x$, if $x = \overline{x_{n-1} \ldots x_0}$ in binary.

Each of the next r lines contains two integers z and g ($0 \le z \le 2^n - 1$, $0 \le g \le 10^9$). If $z = \overline{z_{n-1} \dots z_0}$ in binary, then this means to set $f(z_0, \dots, z_{n-1}) \to g$.

Output

Print $r+1$ lines, the i-th of which denotes the value of the game f during the i-th round. Your answer must have absolute or relative error within 10^{-6}.

Formally, let your answer be a, and the jury's answer be b. Your answer is considered correct if $\frac{|a - b|}{\max{(1, |b|)}} \le 10^{-6}$.

Examples

input

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

output

```
1.500000
2.250000
3.250000
```

input

```
1 0
2 3
```

output

```
2.500000
```

input

```
2 0
1 1 1 1
```

output

```
1.000000
```

Note

Consider the second test case. If Allen goes first, he will set $x_1 \to 1$, so the final value will be 3. If Bessie goes first, then she will set $x_1 \to 0$ so the final value will be 2. Thus the answer is 2.5.

In the third test case, the game value will always be 1 regardless of Allen and Bessie's play.