F. Design Tutorial: Change the Goal

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

There are some tasks which have the following structure: you are given a model, and you can do some operations, you should use these operations to achive the goal. One way to create a new task is to use the same model and same operations, but change the goal.

Let's have a try. I have created the following task for Topcoder SRM 557 Div1-Hard: you are given n integers $x_1, x_2, ..., x_n$. You are allowed to perform the assignments (as many as you want) of the following form x_i ^= x_j (in the original task i and j must be different, but in this task we allow i to equal j). The goal is to maximize the sum of all x_i.

Now we just change the goal. You are also given n integers $y_1, y_2, ..., y_n$. You should make $x_1, x_2, ..., x_n$ exactly equal to $y_1, y_2, ..., y_n$. In other words, for each i number x_i should be equal to y_i.

Input

The first line contains an integer n $(1 \le n \le 10000)$. The second line contains n integers: x_1 to x_n $(0 \le x_i \le 10^9)$. The third line contains n integers: y_1 to y_n $(0 \le y_i \le 10^9)$.

Output

If there is no solution, output `-1`.

If there is a solution, then in the first line output an integer m $(0 \le m \le 1000000)$ – the number of assignments you need to perform. Then print m lines, each line should contain two integers i and j $(1 \le i, j \le n)$, which denote assignment x_i ^= x_j.

If there are multiple solutions you can print any of them. We can prove that under these constraints if there exists a solution then there always exists a solution with no more than 10^6 operations.

Examples

input
2 3 5 6 0
output
2 1 2 2 2

input
5 0 0 0 0 0 1 2 3 4 5
output
-1

input
3 4 5 6 1 2 3
output

```
5
3 1
1 2
2 2
2 3
3 1
```

input

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

output

```
-1
```

Note

Assignment a ^= b denotes assignment a = a ^ b, where operation "^" is bitwise XOR of two integers.