

IOI Training Camp 2018 Final Test 3

Add Rotate XOR

You are given a permutation P of the integers $\{0, 1, 2, \dots, 254, 255\}$. You have an array which is initially $(0, 1, 2, \dots, 254, 255)$. You want to transform this into the permutation P . To do so, you are allowed three types of operations:

- Type 0: You can add any integer x ($0 \leq x \leq 255$) to each and every element of the array. This addition takes place modulo 256.
- Type 1: Consider each element of the array as an 8-bit number. You can take any integer x ($0 \leq x < 8$) and cyclically rotate the bits of each and every element of the array left by x bits.
- Type 2: You can take any integer x ($0 \leq x \leq 255$) and XOR each and every element of the array with x .

You need to find a sequence of operations to get to P .

Input

The first line of the input contains 256 integers, which represent the permutation P .

Output

The first line should contain a single integer, K , which is the number of operations you are going to perform. The i -th of the next K lines should contain two integers, t and x , which denote that this is a type t operation with this value of x .

Subtasks

Subtask 1 (35 Points):

- There will be only one testfile in this subtask, and that will contain “1 0 2 3 4 ... 255” as the input.

Subtask 2 (65 Points):

- There will be multiple test files, and your score will be the minimum score among all these testfiles. Your score in a single testfile is: $165 - 30 * \log_{10} K$, where K is the number of operations you have performed. Your score will be 0 if your sequence of operations is not valid. Furthermore, if you solve the testfile in less than or equal to the number of steps that the author's solution takes, you get the full 65 points for that file.

Sample Input 1

We will take a permutation of 0 to 2 as an example and assume that the initial array is $(0, 1, 2)$.

```
1 2 0
```

Sample Output 1

This is not a correct answer even for the sample input. This is only for illustrative purposes.

```
3
2 2
1 7
0 254
```

Explanation

Initial array is (0, 1, 2). In binary form, represented as 8-bit, it would be (00000000, 00000001, 00000010).

The first operation XORs each element with 2, which is 00000010. So the array becomes (00000010, 00000011, 00000000).

The second operation rotates every number's binary form cyclically, 7 times to the left. So the array becomes (00000001, 10000001, 00000000).

The third operation adds 254 to each element modulo 256. So the array becomes (11111111, 01111111, 11111110).

Limits

Time: 2 seconds

Memory: 512 MB