

## Triple Hunting

In this somewhat silly problem, you must write a program that reads in a list of non-negative integers and identifies all the ‘triples’ - the multiples of three. For the purposes of this problem, a triple is defined as *any* number that can be expressed in the form  $(3 * integer)$ . The numbers 0 and 3 are also considered to be triples.

To make things a little more complicated, once you identify all the triples you will need to output their locations in the original list.

### Input

The first line of the input will consist of the single integer  $n$ , the number of integers in the list. ( $1 \leq n \leq 50,000$ ) Following this will be  $n$  lines describing the list, each containing a single integer between 0 and 2,000,000,000.

### Output

If there were any multiples of three in the list, the first line of your output should contain the single integer  $k$ , the number of triples found.

The second line should contain  $k$  space-separated integers, describing the positions of the triples, in ascending order. For example, if the 5-th number in the list is a triple, you should output 5.

If there are no triples in the list at all, you are instead to output **Nothing here!** on a single line.

### Sample Input 1

```
7
10
12
8
9
3
29
30
```

### Sample Output 1

```
4
2 4 5 7
```

### Sample Input 2

```
7
16
49
2
10
28
55
31
```

### Sample Output 2

```
Nothing here!
```

**Explanation**

In the first example, four of the integers in the list are triples: the 2nd, 4th, 5th and 7th. (Their values are  $12 = 3 \times 4$ ,  $9 = 3 \times 3$ ,  $3 = 3 \times 1$ , and  $30 = 3 \times 10$ .) In the second example there are no triples in the list.