Making Change

Author

Bruce Merry

Introduction

The cows have decided that they are tired of the strange American system of coins, being particularly annoyed by the large gap between the quarter ($0.25) and the $1 coins that force one to carry lots of quarters. They are going to create their own system of coins with sensible values.

However one feature of the American system that they like is that one can always make up a value with a "greedy" algorithm and get the minimal number of coins. The greedy algorithm says that one should always pick the largest value coin that is not greater than the amount still needed. So for example to make 83c in the US system (which has coins of values 1c, 2c, 5c, 10c, 25c and $1), one first picks three 25c coins to make 75c, then a 5c to make 80c, then a 2c and finally a 1c. This results in six coins, and there is no way to make 83c with fewer coins. However if the coins had been 1c, 4c and 6c, then making 8c using a greedy algorithm takes 3 coins (6+1+1) even though it is possible to make 8c using 2 coins (4+4).

Task

You must write a program to determine whether the cow's proposed set of coins satisfies the greedy property, and if not compute the smallest value for which the greedy algorithm is not optimal.

Input (change.in)

The first line is N, the number of coins. Each of the next N lines contains the value of a coin, in cents. Values may be repeated (e.g. for special editions). It is guaranteed that at least one coin will have the value of 1c.

Sample input

```
3
1
6
4
```

Output (change.out)

If the coin set does not have the greedy property, output the smallest value for which the greedy algorithm is non-optimal, followed by the minimum number of coins required to create this value (space separated).

If the coin set has the greedy property, output "−1 −1".

Sample output

```
8 2
```

Constraints

- $1 \le N \le 100$
- $1 \le$ each coin ≤ 10000

50% constraints

- $1 \le N \le 10$

Time limit

2 seconds.

Scoring

An incorrectly formatted output file scores 0.

If the coin set has the greedy property, a correct answer scores 100% and an incorrect answer scores 0.

If the coin set does not have the greedy property, then 50% is awarded for getting the first output correct. If (and only if) this output is correct, 50% is awarded for getting the second field correct.