

# Caper

A series of  $n$  integers is called a *caper* if the absolute values of the differences between the consecutive values from the sequence assume all the values between 1 and  $n - 1$ . For example, the series

```
1 4 2 3
```

is a caper, because the absolute differences are respectively equal to 3, 2 and 1. The definition states that each sequence which consists of one single integer is a caper. Write a program that determines whether a given set of integers is a caper or not.

## Assignment

Write a function `caper` that takes a list of integers as argument and returns a Boolean value as a result. If the list of numbers that is passed as an argument is a caper, this function returns the value `True`. Otherwise, the function returns the value `False`.

## Example

```
>>> caper([1, 4, 2, 3])
True
>>> caper([1, 4, 2, 3, -1, 6])
False
```

Een reeks van  $n$  gehele getallen wordt een *bokkesprong* genoemd, indien de absolute waarden van de verschillen van opeenvolgende waarden uit de reeks alle waarden tussen 1 en  $n - 1$  aannemen. Zo is bijvoorbeeld de reeks

```
1 4 2 3
```

een bokkesprong, omdat de absolute verschillen respectievelijk gelijk zijn aan 3, 2 en 1. De definitie houdt in dat elke reeks die bestaat uit één enkel geheel getal een bokkesprong is. Gevraagd wordt om een programma te schrijven dat bepaalt of een gegeven reeks gehele getallen een bokkesprong is of niet.

## Opgave

Schrijf een functie `bokkesprong` die een lijst van gehele getallen als argument neemt en die een Booleaanse waarde als resultaat teruggeeft. Als de lijst van getallen die als argument doorgegeven wordt een bokkesprong is, geeft deze functie de waarde `True` terug. Anders geeft de functie de waarde `False` terug.

## Voorbeeld

```
>>> bokkesprong([1, 4, 2, 3])
True
>>> bokkesprong([1, 4, 2, 3, -1, 6])
False
```