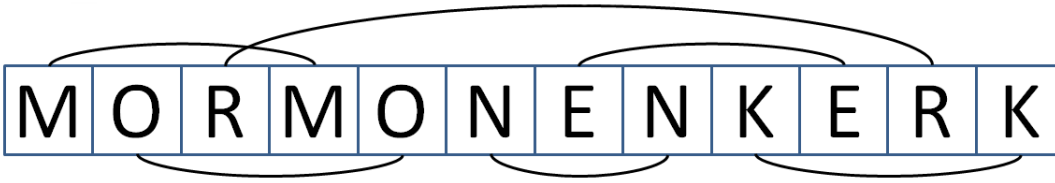


Caucasus

We say a word is *balanced* if all letters occur the same number of times. For example, the word *Kaukasus* counts four pairs of letters, the word *MORMONENKERK* counts six pairs.



Assignment

- Write a function `letterCount` to which a string must be given. The function must print a dictionary that portrays every letter that occurs in the given string on the amount of times that that letter occurs in the string. All characters of the word that are not a letter, must be generated. On counting the number of occurrences of a letters, no distinction may be made between lowercase and uppercase letters, and the representation as lowercase letter must be used as key in the dictionary that is printed.
- Write a function `balanced` to which a string must be given. The function must print a Boolean value, that indicates whether every letter of the word occurs a set number of times n . To this applies that $n \geq 2$. Counting the number of occurrences of the letters in a word must happen in the same way as with the function `letterCount`.

Example

```
>>> letterCount('Caucasus')
{'a': 2, 'c': 2, 'u': 2, 's': 2}
>>> letterCount('teammate')
{'a': 2, 'm': 2, 'e': 2, 't': 2}
>>> letterCount('SCINTILLESCEMENT')
{'c': 2, 'e': 2, 'i': 2, 'l': 2, 'n': 2, 's': 2, 't': 2}
>>> letterCount('chachacha')
{'a': 3, 'h': 3, 'c': 3}
>>> letterCount('blahblahblah')
{'a': 4, 'h': 4, 'b': 4, 'l': 4}
>>> letterCount('intestine')
{'i': 2, 's': 1, 'e': 2, 't': 2, 'n': 2}

>>> balanced('Caucasus')
True
>>> balanced('teammate')
True
>>> balanced('SCINTILLESCEMENT')
True
>>> balanced('lysosome')
False
>>> balanced('intestines')
True
>>> balanced('mesosome')
True
```

We zeggen dat een woord *evenwichtig* is als alle letters even vaak voorkomen. Het woord

Kaukasus telt bijvoorbeeld vier paar letters, het woord MORMONENKERK telt er zes paar.



Opgave

- Schrijf een functie `aantalLetters` waaraan een string moet doorgegeven worden. De functie moet een dictionary teruggeven die elke letter die in de gegeven string voorkomt afbeeldt op het aantal keer dat die letter in de string voorkomt. Alle karakters van het woord die geen letter zijn, moeten hierbij genegeerd worden. Bij het tellen van het aantal voorkomens van een letter mag geen onderscheid gemaakt worden tussen kleine letters en hoofdletters, en de voorstelling als kleine letter moet als sleutel gebruikt worden in de dictionary die wordt teruggegeven.
- Schrijf een functie `evenwichtig` waaraan een string moet doorgegeven worden. De functie moet een Booleaanse waarde teruggeven, die aangeeft of elke letter in het woord een vast aantal keer n voorkomt. Hierbij moet gelden dat $n \geq 2$. Het tellen van het aantal voorkomens van de letters in het woord moet op dezelfde manier gebeuren als bij de functie `aantalLetters`.

Voorbeeld

```
>>> aantalLetters('Kaukasus')
{'a': 2, 'k': 2, 'u': 2, 's': 2}
>>> aantalLetters('wederwoord')
{'o': 2, 'r': 2, 'e': 2, 'd': 2, 'w': 2}
>>> aantalLetters('MORMONENKERK')
{'e': 2, 'k': 2, 'm': 2, 'o': 2, 'n': 2, 'r': 2}
>>> aantalLetters('chachacha')
{'a': 3, 'h': 3, 'c': 3}
>>> aantalLetters('blablabla')
{'a': 4, 'b': 4, 'l': 4}
>>> aantalLetters('stroommeter')
{'e': 2, 'm': 2, 'o': 2, 's': 1, 'r': 2, 't': 2}

>>> evenwichtig('Kaukasus')
True
>>> evenwichtig('wederwoord')
True
>>> evenwichtig('MORMONENKERK')
True
>>> evenwichtig('chachacha')
True
>>> evenwichtig('blablabla')
True
>>> evenwichtig('stroommeter')
False
```