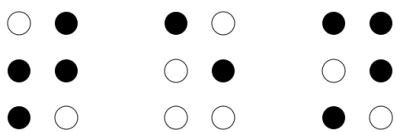


Self-fulfillers

A self-fulfiller (also called **autological word**) is a word or an expression that expresses a property that it also possesses. For example, the word `short` is short, `noun` is a noun, `English` is English, `pentasyllabic` has five syllables, `word` is a word, and `sesquipedalian` is a long word. The word or expression thus speaks for itself because it self-references one of its own properties.

In 2011 Dave Morice observed that the word `LUCK` requires 7 penstrokes, `BLACKJACK` requires 21 penstrokes, `FREEZING POINT` requires 32 penstrokes and `HOURS IN A DAY` requires 24 penstrokes. But there are other possibilities to assign integer values to each letter in the alphabet that give a special or literal meaning to particular words or expressions. We list a few examples.

`TEN` is spelled with ten raised dots in Braille.



The word `TEN` in Braille.

`FIFTEEN` is spelled with 15 dots and dashes in International Morse Code.



The word `FIFTEEN` in morse code.

`TWELVE` is worth 12 points in Scrabble.



The word `TWELVE` in Scrabble.

Assignment

- Write a function `lettervalue` that takes a list or tuple containing 26 integers. Each integer corresponds to the letter at the same position in the alphabet. The function must return a dictionary that maps each uppercase letter in the alphabet to its corresponding integer value in the given list or tuple.
- Use the function `lettervalue` to write a function `wordvalue` that takes two arguments: a string and a list or tuple containing 26 integers. The second argument has the same meaning as the argument passed to the function `lettervalue`. The function must return the integer that results as the sum of the values of the individual letters in the given string. In computing this sum, the function must treat the given string in a case insensitive way and ignore all characters that are not letters.

Example

```
>>> penstrokes = [3, 3, 1, 2, 4, 3, 2, 3, 1, 2, 3, 2, 4, 3, 1, 2, 2, 3, 1, 2, 1, 2, 4, 2, 3, 3]
```

```

>>> value = lettervalue(penstrokes)
>>> value['L']
2
>>> value['U']
1
>>> value['C']
1
>>> value['K']
3

>>> wordvalue('LUCK', penstrokes)
7
>>> wordvalue('blackjack', penstrokes)
21
>>> wordvalue('FREEZING POINT', penstrokes)
32
>>> wordvalue('Hours in a Day', penstrokes)
24
>>> wordvalue('Twenty-nine', penstrokes)
29

>>> braille = (1, 2, 2, 3, 2, 3, 4, 3, 2, 3, 2, 3, 3, 4, 3, 4, 5, 4, 3, 4, 3, 4, 4, 4, 5, 4)
>>> wordvalue('TEN', braille)
10
>>> wordvalue('ten + ten', braille)
20

>>> morse = [2, 4, 4, 3, 1, 4, 3, 4, 2, 4, 3, 4, 2, 2, 3, 4, 4, 3, 3, 1, 3, 4, 3, 4, 4, 4]
>>> wordvalue('fifteen', morse)
15
>>> wordvalue('Se7en', morse)
7

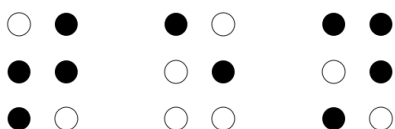
>>> scrabble = (1, 3, 3, 2, 1, 4, 2, 4, 1, 8, 5, 1, 3, 1, 1, 3, 10, 1, 1, 1, 1, 4, 4, 8, 4, 10)
>>> wordvalue('TWELVE', scrabble)
12
>>> wordvalue('Life, Universe and Everything', scrabble)
42

```

Een **zelfvervuller** (ook wel **autologie** genoemd) is een woord dat of uitdrukking die op zichzelf betrekking heeft. De woorden of uitdrukkingen spreken dus voor zichzelf omdat ze verwijzen naar iets dat mede op henzelf betrekking heeft, waar ze als zodanig (door hun vorm) mee te maken hebben.

Zo heb je bijvoorbeeld 7 pennestreken nodig om het woord LUCK te schrijven, 21 om BLACKJACK te schrijven, 32 om FREEZING POINT te schrijven en 24 om HOURS IN A DAY te schrijven. Maar er zijn nog andere mogelijkheden om aan elke letter van het alfabet een waarde toe te kennen, waardoor woorden of uitdrukkingen een speciale of een letterlijke betekenis krijgen. Enkele voorbeelden.

TEN wordt geschreven met 10 verhoogde punten in Braille.



Het woord TEN in Braille.

FIFTEEN wordt geschreven met 15 punten en strepen in de internationale morsecode.

····· ·· ····· - · · -·

Het woord FIFTEEN in morsecode.

TWELVE levert 12 punten op in Scrabble.



Het woord TWELVE in Scrabble.

Opgave

- Schrijf een functie `letterwaarde` waaraan een lijst of een tuple met 26 gehele getallen moet doorgegeven worden. Elk van deze getallen correspondeert met de hoofdletter op de overeenkomstige positie in het alfabet. De functie moet een dictionary teruggeven die elke hoofdletter uit het alfabet afbeeldt op zijn corresponderende waarde in de gegeven lijst of het gegeven tuple.
- Gebruik de functie `letterwaarde` om een functie `woordwaarde` te schrijven. Aan deze functie moeten twee argumenten doorgegeven worden: een string en een lijst of een tuple met 26 gehele getallen. Het tweede argument heeft dezelfde betekenis als het argument dat aan de functie `letterwaarde` wordt doorgegeven. De functie moet als resultaat een geheel getal teruggeven dat de som is van de waarden van de individuele letters in de gegeven string. Hierbij mag geen onderscheid gemaakt worden tussen hoofdletters en kleine letters, en moeten alle karakters die geen letter zijn genegeerd worden.

Voorbeeld

```
>>> pennestreken = [3, 3, 1, 2, 4, 3, 2, 3, 1, 2, 3, 2, 4, 3, 1, 2, 2, 3, 1, 2, 1, 2, 4, 2, 3, 3]
>>> waarde = letterwaarde(pennestreken)
>>> waarde['L']
2
>>> waarde['U']
1
>>> waarde['C']
1
>>> waarde['K']
3

>>> woordwaarde('LUCK', pennestreken)
7
>>> woordwaarde('blackjack', pennestreken)
21
>>> woordwaarde('FREEZING POINT', pennestreken)
32
>>> woordwaarde('Hours in a Day', pennestreken)
24
>>> woordwaarde('Twenty-nine', pennestreken)
29

>>> braille = (1, 2, 2, 3, 2, 3, 4, 3, 2, 3, 2, 3, 3, 4, 3, 4, 5, 4, 3, 4, 3, 4, 4, 4, 5, 4)
```

```
>>> woordwaarde('TEN', braille)
```

```
10
```

```
>>> woordwaarde('ten + ten', braille)
```

```
20
```

```
>>> morse = [2, 4, 4, 3, 1, 4, 3, 4, 2, 4, 3, 4, 2, 2, 3, 4, 4, 3, 3, 1, 3, 4, 3, 4, 4, 4]
```

```
>>> woordwaarde('fifteen', morse)
```

```
15
```

```
>>> woordwaarde('Se7en', morse)
```

```
7
```

```
>>> scrabble = (1, 3, 3, 2, 1, 4, 2, 4, 1, 8, 5, 1, 3, 1, 1, 3, 10, 1, 1, 1, 1, 4, 4, 8, 4, 10)
```

```
>>> woordwaarde('TWELVE', scrabble)
```

```
12
```

```
>>> woordwaarde('Life, Universe and Everything', scrabble)
```

```
42
```