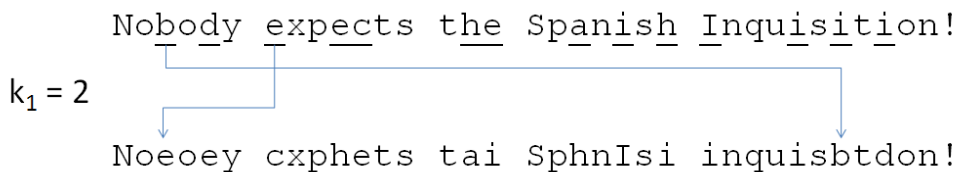


Triple rotation

To convert a message into unreadable gibberish, *triple rotation* can be used. The triple rotation uses three integers k_1 , k_2 and k_3 which are the key to the cipher. To encrypt a given message, the characters of the message are divided into three groups: the letters a to i form the first group (both uppercase and lowercase), the letters j to r the second group (uppercase and lowercase), and all other characters (including numbers, spaces and punctuation) form the third group. Within group i ($i = 1, 2, 3$) all characters thereafter are shifted k_i positions to the left, with the characters that drop out on the left being placed on the right. Every group is thus rotated independently from the other two groups. The decoding then takes place on the basis of a rotation with k_i positions to the right within each group i ($i = 1, 2, 3$).

If, for example, we want to encrypt the message "Nobody expects the Spanish Inquisition!" with the triple rotation with $k_1 = 2$, $k_2 = 3$ and $k_3 = 1$. For this message group 1 is formed with the letters b, d, e, e, c, h, e, a, i, h, l, i, i, i which occur at positions 3, 5, 8, 11, 12, 17, 18, 22, 24, 26, 28, 32, 34, 36 in the message. We move all letters of this group $k_1 = 2$ positions to the left, with the letters b and d, which fall out, being placed at the end of the sequence. For group 1 this results in the new series of letters e, e, c, h, e, a, i, h, l, i, i, i, b, d, which are then put back in the original positions of group 1. This step of the procedure is illustrated in the figure below.



triple rotation

When this procedure is executed for group 2 with $k_2 = 3$ and for group 3 with $k_3 = 1$, then the encrypted message reads as follows "ppene xctnhes t aiSsqhol iuinNsitb!dooy". Try making head or tail of that.

Assignment

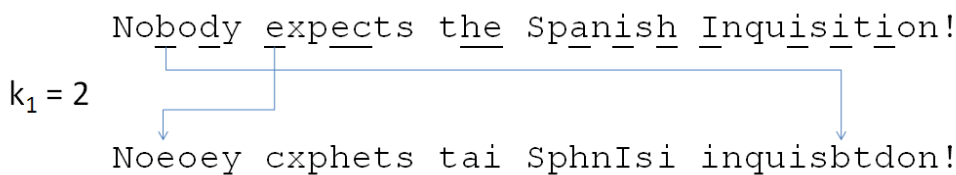
- Write a function `encode` that can be used to encode a given a sentence according to the principle of triple rotation. Four arguments must be passed to this function: a sentence, and the values k_1 , k_2 and k_3 that must be used for the triple rotation. The function must return the encoded sentence as a result.
- Write a function `decode` that can be used for decode a sentence that was encrypted according to the principle of triple rotation. Four arguments must be passed to this function: a given encoded sentence and values k_1 , k_2 and k_3 , which were used for the triple rotation. The function must return the original sentence as a result.

Example

```
>>> encode('Nobody expects the Spanish Inquisition!', 2, 3, 1)
'ppene xctnhes t aiSsqhol iuinNsitb!dooy'
>>> decode('ppene xctnhes t aiSsqhol iuinNsitb!dooy', 2, 3, 1)
'Nobody expects the Spanish Inquisition!'
```

Om een boodschap om te zetten in onleesbare wartaal kan gebruik gemaakt worden van een codeertechniek die *triplerotatie* genoemd wordt. De triplerotatie maakt gebruik van drie natuurlijke getallen k_1 , k_2 en k_3 die de sleutel tot het geheimschrift vormen. Om een gegeven boodschap te coderen, worden de karakters van de boodschap onderverdeeld in drie groepen: de letters a tot en met i vormen de eerste groep (zowel hoofdletters als kleine letters), de letters j tot en met r de tweede groep (opnieuw zowel hoofdletters als kleine letters), en alle andere karakters (inclusief cijfers, spaties en leestekens) vormen de derde groep. Binnen groep i ($i = 1, 2, 3$) worden alle karakters daarna k_i posities naar *links* opgeschoven, waarbij de karakters die links uitvallen aan de rechterkant geplaatst worden. Hierbij wordt elke groep dus geroteerd onafhankelijk van de andere twee groepen. Het decoderen gebeurt dan op basis van een rotatie met k_i posities naar *rechts* binnen elke groep i ($i = 1, 2, 3$).

Neem bijvoorbeeld de boodschap "Nobody expects the Spanish Inquisition!" die we willen coderen volgens de triplerotatie met $k_1 = 2$, $k_2 = 3$ en $k_3 = 1$. Voor deze boodschap wordt groep 1 gevormd door de reeks letters b, d, e, e, c, h, e, a, i, h, l, i, i, i die voorkomen op posities 3, 5, 8, 11, 12, 17, 18, 22, 24, 26, 28, 32, 34, 36 in de boodschap. We schuiven alle letters van deze groep $k_1 = 2$ posities naar links, waarbij de letters b en d die links uitgeschoven worden achteraan de reeks worden toegevoegd. Dit resulteert voor groep 1 in de nieuwe letterreeks e, e, c, h, e, a, i, h, l, i, i, i, b, d, die dan terug op de originele posities van groep 1 geplaatst worden. Deze stap van de procedure wordt geïllustreerd in onderstaande figuur.



triplerotatie

Wanneer dezelfde procedure wordt uitgevoerd voor groep 2 met $k_2 = 3$ en voor groep 3 met $k_3 = 1$, dan leest de gecodeerde boodschap als "ppene xctnhes t aiSsqhol iuinNsitb!dooy". Krijg daar maar eens kop of staart aan.

Opgave

- Schrijf een functie `codeer` die kan gebruikt worden om een gegeven zin te coderen volgens het principe van de triplerotatie. Aan deze functie moeten vier argumenten doorgegeven worden: een zin, en de waarden k_1 , k_2 en k_3 die gebruikt moeten worden bij de triplerotatie. De functie moet de gecodeerde zin als resultaat teruggeven.
- Schrijf een functie `decodeer` die kan gebruikt worden om een zin die gecodeerd werd volgens het principe van de triplerotatie terug te decoderen. Aan deze functie moeten vier argumenten doorgegeven worden: een gecodeerde zin, en de waarden k_1 , k_2 en k_3 die gebruikt werden voor de triplerotatie. De functie moet de originele zin als resultaat teruggeven.

Voorbeeld

```
>>> codeer("Nobody expects the Spanish Inquisition!", 2, 3, 1)
'ppene xctnhes t aiSsqhol iuinNsitb!dooy'
>>> decodeer('ppene xctnhes t aiSsqhol iuinNsitb!dooy', 2, 3, 1)
'Nobody expects the Spanish Inquisition!'
```

