



Ciclo 1 Fundamentos de programación

Reto 3 P59

Descripción del problema: Palabras largas

A veces, algunas palabras como "esternocleidomastoideo" o "otorrinolaringólogo" son tan largas que escribirlas muchas veces en un texto es bastante tedioso.

Las diez palabras más largas en español:

- 1-Electroencefalografista,
- 2-Esternocleidomastoideo,
- 3-Electroencefalografía,
- 4-Electroencefalograma,
- 5-Otorrinolaringólogo, 6-Electrocardiograma,
- 7-Electrodoméstico, 8-Arteriosclerosis,
- 9-Paralelepípedo y 10-Caleidoscopio.

Consideremos una palabra demasiado larga, si su longitud es estrictamente superior a 10 caracteres. Todas las palabras demasiado largas deben reemplazarse por una abreviatura especial.

Esta abreviatura se hace así: escribimos la primera y la última letra de una palabra y entre ellas escribimos el número de letras entre la primera y la última letra. Ese número está en sistema decimal y no contiene ceros a la izquierda.

Por lo tanto, "Esternocleidomastoideo" se escribirá como "e20o" y "otorrinolaringólogo" se escribirá como "o17o".

Se sugiere automatizar el proceso de cambio de palabras con abreviaturas. En ese caso, las palabras demasiado largas deben reemplazarse por la abreviatura y las palabras que no son demasiado largas no deben sufrir ningún cambio.



Aporte

Todas las palabras constan de letras latinas minúsculas o mayúsculas y poseen una longitud de 1 a 100 caracteres.

De acuerdo con lo anterior, se requiere que escriba una función que reciba como parámetro: una lista de palabras y que con la ayuda de ciclos la salida muestre o devuelva las palabras abreviadas dentro de una lista así:

Ejemplo:

lista=['word','localization','internationalization','pneumonoultramicroscopicsilicovolcanoc
oniosis']

Entradas:

Nombre	Tipo	Descripción
lista	List[Str]	lista representa una lista de palabras de 1 a 100 caracteres

Salidas:

Tipo del retorno	Descripción
List[Str]	La salida es una lista con las palabras abreviadas, si la palabras tiene 10 caracteres o menos se debe mostrar la misma cadena, si la palabra de entrada tiene mas de 10 caracteres, por ejemplo, localizacion, la salida debe ser "l10n" (le-diez-ene), donde l es la primera letra de la palabra, el 10 (diez) es la cantidad de caracteres intermedios y la n (ene) la letra final.



Esqueleto:

```
def palabras(lista)->list:
```

Nota: En la plataforma debe subirse una función con **el mismo nombre, la misma cantidad de argumentos, y el retorno debe tener exactamente la estructura de la salida que se presenta en este documento**, tener precaución con el uso de tildes y caracteres especiales como la ñ.

Ejemplos:

Entradas:

```
lista = ['oro','localizacion','internationalizacion','electroencefalograma']
```

Salidas:

```
Salida = ['oro','l10n','i18n','e18a']
```

Casos de prueba:

```
print(palabras(['internacionalizacion', 'esternocleido','oro']))  
print(palabras(['internacionalizacion', 'esternocleido','oro','Arteriosclerosis', 'Electroencefalografista','Inconstitucionalmente']))  
print(palabras(['adrenomieloneuropatía', 'Hipopotomonstrosesquipedaliofobia','murcielago']))
```

```
['i18n', 'e11o', 'oro']  
['i18n', 'e11o', 'oro', 'A14s', 'E21a', 'I19e']  
['a19a', 'H31a', 'murcielago']
```