

Introducción a la Programación

Augusto González Omahen

Primer Semestre de 2022

Clase de repaso (*continuación*)

Ejercicio 1

Recordando un poco el algoritmo y su invariante de la clase pasada, vamos a demostrar que esa implementación es correcta respecto de su especificación.

```
1  def sumar_digitos(s:str) -> int:
2      '''
3      Requiere: s solo tiene digitos
4      Devuelve: la suma de los digitos en s
5      '''
6      res:int = 0
7      i:int = 0
8      #A
9      while i < len(s):
10         #B
11         res = res + int(s[i])
12         i = i + 1
13         #C
14     #D
15     return res
```

Invariante

$$0 \leq i \leq \text{len}(s)$$

res tiene la suma de los dígitos de *s* hasta la pos *i* (sin incluir)

Ejercicio 2

En cualquier palabra en español, si a continuación de cada aparición de una vocal se agrega primero una p y luego la misma vocal en minúscula, se obtiene la palabra en el idioma **jeringoso**.

Por ejemplo, *Elefante* es *Epelepenfapantepe*, *Tomate* es *Topomapatepe* y *Digital* es *Dipigipitapal*.

Ejercicio 2

En cualquier palabra en español, si a continuación de cada aparición de una vocal se agrega primero una p y luego la misma vocal en minúscula, se obtiene la palabra en el idioma **jeringoso**.

Por ejemplo, *Elefante* es *Epelepenfapantepe*, *Tomate* es *Topomapatepe* y *Digital* es *Dipigipitapal*.

Sea la siguiente especificación:

Traduce un texto sin tildes de español a jeringoso.

Encabezado: `traducir_a_jeringoso(texto:str) → str`

Requiere: texto es una oración en español sin tildes.

Devuelve: la traducción de texto a jeringoso.

- (a) Dada la implementación de la siguiente pagina, proponer un invariante.
- (b) Dado el invariante, demostrar que la implementación es correcta respecto de su especificación.
- (c) (para hacer en casa) Descargar el archivo **traductor_jeringoso_testing.py** y correr los tests sobre la implementación propuesta. En caso de encontrarse errores, corregir los test hasta que pasen todos.

Ejercicio 2

```
1  def traducir_a_jeringoso(texto:str) ->
    int:
2      '''
3      Requiere: texto es en español y sin
        tildes
4      Devuelve: la traducción de texto a
        jeringoso
5      '''
6      res:str = ''
7      i:int = 0
8      #A
9      while i < len(texto):
10         #B
11         letra_actual:str = texto[i]
12         res = res + letra_actual
13         if es_vocal(letra_actual):
14             res = res+'p'+letra_actual.lower()
15         i = i + 1
16         #C
17     #D
18     return res
```

Ejercicio 2

```
1  def traducir_a_jeringoso(texto:str) ->
    int:
2      '''
3      Requiere: texto es en español y sin
        tildes
4      Devuelve: la traducción de texto a
        jeringoso
5      '''
6      res:str = ''
7      i:int = 0
8      #A
9      while i < len(texto):
10         #B
11         letra_actual:str = texto[i]
12         res = res + letra_actual
13         if es_vocal(letra_actual):
14             res = res+'p'+letra_actual.lower()
15         i = i + 1
16         #C
17     #D
18     return res
```

Invariante

$0 \leq i \leq \text{len}(\text{texto})$

res tiene la traducción a jeringoso de texto hasta la pos **i** (sin incluir)

Resumendo

Cosas importantes:

1. Tener en cuenta que con cada función que tenga ciclo hay que demostrar dos cosas: finalización y correctitud.
2. Hacer una demostración no es contar línea a línea el algoritmo y tampoco son necesarios los ejemplos!