

Ejercicios con secuencias en Python

1. Definir una función que, al recibir una cadena de texto, cuente cuántas vocales hay y devuelva dicho valor.
2. Definir una función que, al recibir una cadena de texto, cuente cuántas palabras hay y devuelva dicho valor.
3. Definir una función que devuelva la suma dos números. Utilizar esa función para sumar tres números.

4. Definir la función

```
# rota : (int, List[A]) -> List[A]
# tal que rota(n, xs) es la lista obtenida poniendo los n primeros
# elementos de xs al final de la lista. Por ejemplo,
# rota(1, [3, 2, 5, 7]) == [2, 5, 7, 3]
# rota(2, [3, 2, 5, 7]) == [5, 7, 3, 2]
# rota(3, [3, 2, 5, 7]) == [7, 3, 2, 5]
```

5. Definir la función

```
# rango : (List[int]) -> List[int]
# tal que rango(xs) es la lista formada por el menor y mayor elemento
# de xs.
# rango([3, 2, 7, 5]) == [2, 7]
```

6. Definir la función

```
# interior : (list[A]) -> list[A]
# tal que interior(xs) es la lista obtenida eliminando los extremos de
# la lista xs. Por ejemplo,
# interior([2, 5, 3, 7, 3]) == [5, 3, 7]
```

7. Definir la función

```
# finales : (int, list[A]) -> list[A]
# tal que finales(n, xs) es la lista formada por los n finales
# elementos de xs. Por ejemplo,
# finales(3, [2, 5, 4, 7, 9, 6]) == [7, 9, 6]
```

8. Ejercicio 13. Definir la función

```
# segmento : (int, int, list[A]) -> list[A]
# tal que segmento(m, n, xs) es la lista de los elementos de xs
# comprendidos entre las posiciones m y n. Por ejemplo,
# segmento(3, 4, [3, 4, 1, 2, 7, 9, 0]) == [1, 2]
# segmento(3, 5, [3, 4, 1, 2, 7, 9, 0]) == [1, 2, 7]
# segmento(5, 3, [3, 4, 1, 2, 7, 9, 0]) == []
# -----
```

9. Definir la función

```
# extremos : (int, list[A]) -> list[A]
# tal que extremos(n, xs) es la lista formada por los n primeros
# elementos de xs y los n finales elementos de xs. Por ejemplo,
# extremos(3, [2, 6, 7, 1, 2, 4, 5, 8, 9, 2, 3]) == [2, 6, 7, 9, 2, 3]
```

10. Las dimensiones de los rectángulos puede representarse
por pares; por ejemplo, (5,3) representa a un rectángulo de base 5 y
altura 3.

Definir la función
mayorRectangulo : (tuple[float, float], tuple[float, float])
-> tuple[float, float]
tal que mayorRectangulo(r1, r2) es el rectángulo de mayor área entre
r1 y r2. Por ejemplo,
mayorRectangulo((4, 6), (3, 7)) == (4, 6)
mayorRectangulo((4, 6), (3, 8)) == (4, 6)
mayorRectangulo((4, 6), (3, 9)) == (3, 9)

11. Definir la función
intercambia : (tuple[A, B]) -> tuple[B, A]
tal que intercambia(p) es el punto obtenido intercambiando las
coordenadas del punto p. Por ejemplo,
intercambia((2,5)) == (5,2)
intercambia((5,2)) == (2,5)