

Introducción a la programación para Ciencia de Datos

Ejercicios.

Para todos ellos hay que realizar varias pruebas que aseguren que funcionan correctamente en todos los casos.

1. Escribe una función `contar_letras(palabra, letra)` que devuelva el número de veces que aparece una letra en una palabra.
2. Escribe una función `eliminar_letras(palabra, letra)` que devuelva una versión de palabra que no contiene el carácter letra.
3. Escribe una función `buscar(palabra, sub)` que devuelva la posición en la que se puede encontrar sub dentro de palabra o -1 en caso de que no esté.
4. Escribe una función `num_vocales(palabra)` que devuelva el número de vocales que aparece en la palabra.
5. Escribe una función `vocales(palabra)` que devuelva las vocales que aparecen en la palabra.
6. Escribe una función `es_inversa(palabra1, palabra2)` que devuelva True si una palabra es la misma que la otra pero con los caracteres en orden inverso. Por ejemplo 'absd' y 'dsba'
7. Escribe una función `comunes(palabra1, palabra2)` que devuelva una cadena formada por los caracteres comunes a las dos palabras.
8. Escribe una función `eco_palabra(palabra)` que devuelva una cadena formada por palabra repetida tantas veces como sea su longitud. Por ejemplo 'hola' -> 'holaholaholahola'
9. Escribe una función `palindromo(frase)` que determine si frase es un palíndromo. Es decir, que se lea igual de izquierda a derecha que de derecha a izquierda (sin considerar espacios).
10. Escribe una función `orden_alfabetico(palabra)` que determine si las letras que forman palabra aparecen en orden alfabético. Por ejemplo: 'abejo'
11. Escribe una función **trocear**(palabra, num) que devuelva una lista con trozos de tamaño num de palabra.
12. Un **anagrama** de una palabra pal1 es una palabra formada con las mismas letras que pal1 pero en orden distinto. Escribe una función `anagrama(palabra1, palabra2)` que determine si es una anagrama. Ejemplo: marta – trama,