

1. Implementa una función que, dado un número entero, devuelva su inverso. Por ejemplo, para $n = 3$, devolvería $1/3 = 0.33$
2. Implementa una función que reciba un entero e imprima por pantalla el número en el que termina con letras. Por ejemplo, para 542 imprimiría por pantalla "dos".
3. Implementa una función que reciba un precio en euros y su IVA, y devuelva el precio del producto IVA incluido.
4. Implementa un programa que calcule el código hash de una cadena de caracteres que reciba como argumento. Para calcular el código hash de una cadena, sumaremos cada valor ASCII de la letra multiplicado por la posición que ocupa. Por ejemplo, para la cadena "hola", su código hash sería $104 \cdot 1 + 111 \cdot 2 + 108 \cdot 3 + 97 \cdot 4 = 1038$.
5. Implementa una función que, dadas dos cadenas de caracteres, devuelva el número de caracteres diferentes entre ellas. Por ejemplo, las cadenas "hola" y "halo", debería devolver 2.
6. Implementa una función que, dada una cadena de caracteres, devuelva el número de vocales que tiene. Por ejemplo, la cadena "prueba" debería devolver 3.
7. Implementa una función que reciba un array de enteros v y un entero n , y devuelva el número de elementos que hay en ese array que son divisibles por n . Por ejemplo, para la entrada $[2,3,5,8,10]$ y 2, debería devolver 3.
8. Implementa la función toUpperCase que, dada una letra recibida como parámetro, la convierta en mayúscula. Por ejemplo, para entrada 'a' debería convertirse en 'A', mientras que la entrada 'B' se mantendría como 'B'.
9. Implementa una función que reciba un float n y un entero r , y modifique n para que se redondee a r decimales. Para ello, basta con multiplicar n por 10^r , almacenarlo en un entero, y asignar a r el resultado de $n / 10^r$.
10. Implementa una función que censure una contraseña recibida como una cadena de caracteres. Para ello, sustituirá cada valor de la contraseña por un carácter '*'.