

PRÁCTICA 1: Criptografía Clásica

RELACIÓN DE EJERCICIOS:

1. Dado el siguiente código Python, que implementa el cifrado Cesar (+3) para el alfabeto inglés en Mayúsculas ($C: M \rightarrow M + 3 \pmod{26}$),

```
def cifradoCesarAlfabetoInglesMAY(cadena):  
    """Devuelve un cifrado Cesar tradicional (+3)"""  
    # Definir la nueva cadena resultado  
    resultado = ''  
    # Realizar el "cifrado", sabiendo que A = 65, Z = 90, a = 97, z = 122  
    i = 0  
    while i < len(cadena):  
        # Recoge el caracter a cifrar  
        ordenClaro = ord(cadena[i])  
        ordenCifrado = 0  
        # Cambia el caracter a cifrar  
        if (ordenClaro >= 65 and ordenClaro <= 90):  
            ordenCifrado = (((ordenClaro - 65) + 3) % 26) + 65  
        # Añade el caracter cifrado al resultado  
        resultado = resultado + chr(ordenCifrado)  
        i = i + 1  
    # devuelve el resultado  
    return resultado
```

se pide implementar la siguiente funcionalidad:

- a) Implementar la función de descifrado Cesar para alfabeto inglés en mayúsculas, la cual descifre los textos cifrados creados por el código anterior.
- b) Modificar las funciones de cifrado y descifrado, para que soporten tanto letras en mayúsculas (A..Z) como letras en minúsculas (a..z) en el alfabeto Inglés.
- c) Modificar las funciones de cifrado y descifrado, para que soporten el cifrado Cesar generalizado ($C: M \rightarrow M + i \pmod{26}$)

Para realizar este ejercicio, se aconseja disponer de una tabla ASCII, la cual muestra la posición de cada una de las letras del alfabeto inglés. Dicha tabla puede consultarse en

<https://en.wikipedia.org/wiki/ASCII>

<https://elcodigoascii.com.ar/codigos-ascii/letra-a-mayuscula-codigo-ascii-65.html>