Programa 05: Encriptación

21 de noviembre de 2022

Carlos Reyes Rico. 217458353

Estructura de Datos 2.

OBJETIVO

Programa:
1) Utilizar dos métodos de encriptación (cesar, xor, DES, AES, Blowfish, puede usar otro algoritmo de encriptación como base algorítmica)
2) cifrado (encriptación)
3) descifrado (desencriptación)
4) Seleccionar un medio de transmisión (por archivos, por imagen, red)
5) No existe restricción de lenguaje de programación
Nota: si utiliza herramientas de cifrado deberán de ser como elementos secundarios y no protagonizar su programa.

Reporte:

A la plataforma debe de subir su programa y un reporte con impresiones de pantalla del funcionamiento de su programa.
Debe de tener una explicación breve que hace su programa
Debe de explicar cómo organizo su algoritmo de encriptación
El reporte debe de tener el siguiente nombre reporte01-PrimerApellido Segundo Apellido Nombre.pdf
El nombre del programa debe ser prog01-PrimerApellido Segundo Apellido Nombre.cpp o comprimir todo el proyecto en un solo archivo esté nombre (.zip)

REPORTE #5 - ENCRIPTACIÓN

El programa que se nos solicita es uno que realiza encriptación y desencriptación de un texto proporcionado por el usuario, al ser programación libre yo realice mi programa en lenguaje Python además de que programe 3 métodos que son el cifrado de cesar, cifrado de afin y cifrado de vigenere.

El cifrado de cesar se basa en rotaciones del alfabeto dichas rotaciones pueden ser seleccionadas por la persona que lo utiliza mientras que el cifrado de afin se basa por decimales predefinidos y en base a esto es que realiza la codificación del mensaje, por último, el cifrado vigenere utiliza una clave principal por la cual empiezaa realizar el cifrado.

La única similitud entre estos es el uso del alfabeto y que se basan solamente en sustitución de letras por medio de rotaciones ya sea una rotación definida o por medio de una clave.

Explicación del programa

Como ya se mencione se utilizó el lenguaje Python para la programación, para empezar, cree un menú de interacción para seleccionar cual cifrado se desea utilizar, para posteriormente preguntar si va a codificar un mensaje o decodificar un mensaje, todo esto con la intención de mantener un control y orden del cifrado que se utiliza.

Menú

```
should_end = False

while not should_end:

    opc=input("Sleccione que cifrado desea usar:\n 'cesar'\n 'afin'\n 'vigenere'\n R= ")

    if opc=="cesar":

        direction = input("Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:\n")

        if (direction == "codificar" or direction == "decodificar"):

            text = input("Escribe tu mensaje:\n").lower()

            try:

                shift = int(input("Seleccione numero de rotaciones:\n"))

                shift = shift % 26

            except ValueError:

                print("Introduzca solo números como valor de cambio")
```

```
            continue
        caesar(start_text=text, shift_amount=shift, cipher_direction=direction)


elif opc == "afin":
    direction = input("Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:\n")
    if (direction == "codificar" or direction == "decodificar"):
        text = input("Escribe tu mensaje:\n").lower()
        if direction == "codificar":
            result=1
        elif direction == "decodificar":
            result=2
        afin(text,result,direction)


elif opc == "vigenere":
    direction = input("Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:\n")
    if (direction == "codificar" or direction == "decodificar"):
        text = input("Escribe tu mensaje:\n").lower()
        vigenere(text,direction)


else:
    print("Por favor seleccione una opcion valida")
restart = input("Escribe 'sí' si quieres volver a ir. De lo contrario escriba 'no'.\n")
if restart == "no" or restart == "n" or restart == "":
    should_end = True
    print("Adios")
```

```
Sleccione que cifrado desea usar:
 'cesar'
 'afin'
 'vigenere'
 R= cesar

Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:
codificar

Escribe tu mensaje:
Hola mundo

Seleccione numero de rotaciones:
5
El resultado de la codificard es: mtqf rzsit

Escribe 'sí' si quieres volver a ir. De lo contrario escriba 'no'.
```

Cifrado cesar

```
def caesar(start_text, shift_amount, cipher_direction):
    end_text = ""
    if cipher_direction == "decodificar":
        shift_amount *= -1
    for char in start_text:
        if char in alphabet:
            position = alphabet.index(char)
            new_position = position + shift_amount
            end_text += alphabet[new_position]
        else:
            end_text += char
    print(f"El resultado de la {cipher_direction}d es: {end_text}")
```

```
Escribe tu mensaje:
Hola mundo

Seleccione numero de rotaciones:
5
El resultado de la codificard es: mtqf rzsit
```

```
Escribe tu mensaje:
mtqf rzsit

Seleccione numero de rotaciones:
5
El resultado de la decodificard es: hola mundo
```

Cifrado afin

```
def afin(msg,result,direction):
    clave=(5,1)
    if result==1:
        codi=encipher_affine(msg, clave)
    else:
        codi=decipher_affine(msg, clave)
    print(f"El resultado de la {direction}d es: {codi}")
```

```
Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:
codificar

Escribe tu mensaje:
hola mundo
El resultado de la codificard es: KTEBJXOQT
```

```
Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:
decodificar

Escribe tu mensaje:
KTEBJXOQT
El resultado de la decodificard es: HOLAMUNDO
```

Cifrado vigenere

```
def vigenere(mesg,direction):
    codi = ''
    key="platano"

    if direction=="codificar":
        mensaje = [mesg[i:i + len(key)] for i in range(0, len(mesg), len(key))]

        for split in mensaje:
            i=0
            for letter in split:
                numero=(letter_to_index[letter] + letter_to_index[key[i]]) % len(alphabet)
                codi += index_to_letter[numero]
```

```
            i+=1
    else:
        mensaje = [mesg[i:i + len(key)] for i in range(0, len(mesg), len(key))]


        for split in mensaje:
            i=0
            for letter in split:
                numero=(letter_to_index[letter] - letter_to_index[key[i]]) % len(alphabet)
                codi += index_to_letter[numero]
                i+=1


    print(f"El resultado de la {direction}d es: {codi}")
```

```
Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:
codificar

Escribe tu mensaje:
holamundo
El resultado de la codificard es: wzltmhbsz
```

```
Escriba 'codificar' para cifrar, escriba 'decodificar' para descifrar:
decodificar

Escribe tu mensaje:
wzltmhbsz
El resultado de la decodificard es: holamundo
```

Conclusión

Cada cifrado tiene su complejidad y grado de protección, en lo personal me parece mejor el cifrado de vigenere, porque este es más complejo que los anteriores por la cuestión de la palabra clave principal que es definida por el que encripta el mensaje así que esto da la certeza de que será más complejo de descifrar, sin embargo los otros dos también cuentan con su rango de complejidad por lo cual son óptimos si se requiere mandar un mensaje encriptado.