DESARROLLO DE ALGORITMOS CON ESTRUCTURAS DE DATOS

Evaluación

UCO ONLINE

Desarrollo de algoritmos con estructuras de datos

Objetivos

Desarrollar un programa en Python que haga uso de las estructuras de datos explicadas (listas, tuplas, conjuntos, diccionarios)

Temporización

60 minutos

Enunciado

Descripción

Un cifrado es un medio para ocultar un mensaje, donde las letras del mensaje son sustituidas o traspuestas por otras letras, pares de letras y algunas veces por muchas letras. En criptografía el cifrado clásico es un tipo de cifrado que fue usado históricamente pero que ahora ha caído, mayormente, en desuso. En general, los cifrados clásicos operan en un alfabeto de letras (como "A-Z"), a las cuales se les aplican métodos a mano o con aparatos mecánicos muy simples

Objetivo

Implementar un programa que permita cifrar y descifrar un texto de acuerdo a varios métodos de cifrado:

- Cifrado César
- Cifrado con diccionario

Descripción de los método de cifrados

Cifrado César

El cifrado César de un mensaje, dado un mensaje, y una clave privada, *k*, transformará cada letra del mensaje en un número, *m*, (utilizando la Tabla 1). A dicho número se le suma la clave privada k y se hace módulo 32. La codificación de un mensaje se puede expresar como: $c=(m+k) \ mod \ 32$. Después se vuelve a convertir *c* en una letra.

Por ejemplo, si el mensaje es “MAR” y la clave privada es 25, resulta que:

- M=13 → (13+25) mod 32 = 6 → F
- A=1 → (1+25) mod 32 = 26 → Z
- R=18 → (18+25) mod 32 =11 → K

Por lo tanto, el mensaje cifrado, será “FZK”.

Descifrado César

A partir de un mensaje cifrado y la clave privada, *k*, utilizada para cifrarlo, se obtiene el mensaje sin cifrar. Para ello, se convierte cada carácter del mensaje cifrado en un número, *c*, utilizando la Tabla 1 y se calcula *m=(c-k) mod 32*. Después se transforma *m* en el carácter correspondiente.

Por ejemplo, si desciframos el mensaje anterior “FZK” tenemos:

- F=6 → (6-25)=-19 + 32=13 → M
- Z= 26 → (26-25) mod 32 = 1 → A
- K=11 → (11-25) =-14+32=18 → R

*→0	H→8	P→16	X→24
A→1	I→9	Q→17	Y→25
B→2	J→10	R→18	Z→26
C→3	K→11	S→19	(→27
D→4	L→12	T→20	)→28
E→5	M→13	U→21	,→29
F→6	N→14	V→22	?→30
G→7	O→15	W→23	!→31

Tabla 1: Alfabeto a utilizar para el cifrado César

Cifrado con diccionario

El cifrado con diccionario de un mensaje consiste en transformar cada letra del mensaje en su letra cifrada utilizando el diccionario proporcionado.

Por ejemplo, si el mensaje es MAR y el diccionario el de la Tabla 2, el mensaje cifrado será “?B)”:

- M → ?
- A → B
- R →)

Descifrado con diccionario

A partir de un mensaje cifrado y el diccionario utilizado para ello, podemos obtener el mensaje original de la siguiente manera:

1. Crear un diccionario inverso, en el que las claves se convierten en valores y los valore en clave.
2. Aplicar el cifrado pero utilizando el diccionario inverso

Por ejemplo, si desciframos el mensaje anterior "?B)" utilizando el diccionario de la Tabla 2 tendremos:

- ? → M
- B → A
-) → R

*→ J	H→C	P→!	X→U
A→B	I→W	Q→X	Y→P
B→S	J→E	R→)	Z→T
C→Y	K→*	S→V	(→O
D→K	L→(	T→,	)→N
E→A	M→?	U→H	,→G
F→R	N→Z	V→F	?→L
G→M	O→Q	W→D	!→I

Tabla 2: Diccionario para cifrado

Indicaciones

Antes de cifrar o descifrar el mensaje proporcionado el programa deberá realizar el siguiente preprocesado:

- Transformar las minúsculas en mayúsculas.
- Transformar los espacios en blanco en el carácter '*'.
- Transformar los caracteres que no se encuentren dentro del alfabeto en el carácter '*'.

Invocación del script

El programa recibirá 2 o 3 parámetros, según se quiera cifrar/descifrar mediante diccionario o César:

- Mensaje a procesar
- Si se va a cifrar (c) o descifrar (d)
- La clave privada *k* en caso de querer utilizar el cifrado César

Ejemplos

python3 criptografia.py “mar” c 25

Salida → FZK

python3 criptografia.py “fzk” d 25

Salida → MAR

python3 criptografia.py “mar” c

Salida → ?B)

python3 criptografia.py “cat” d

Salida → CEZ