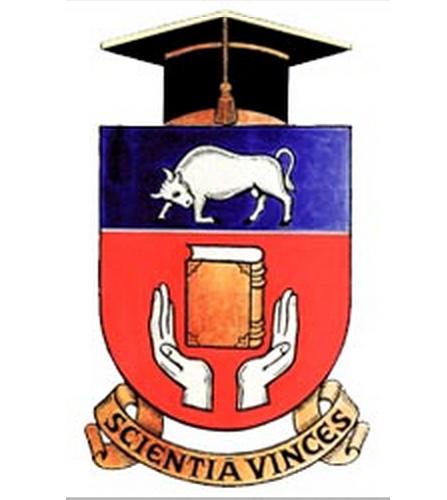
Universitatea de Stat din Tiraspol

Facultatea de Fizica Matematica si Informatica



Cifrul Caesar

**Elaborat de**: Dordea Pavel, student gr.3i

**Verificat**: Globa Angela

**Chișinău, 2021**

**Sistemul Caesar**

***#Despre***

În criptografie, **cifrul lui Cezar**, numit și **cifru cu deplasare**, **codul lui Cezar** sau **deplasarea lui Cezar**, este una dintre cele mai simple și mai cunoscute tehnici de criptare. Este un tip de cifru al substituției, în care fiecare literă din textul inițial este înlocuită cu o literă care se află în alfabet la o distanță fixă față de cea înlocuită. De exemplu, cu o deplasare de cinci poziții în alfabetul limbii române, A este înlocuit cu D, Ă devine E și așa mai departe. Această metodă este numită așa după Iulius Cezar, care o folosea pentru a comunica cu generalii săi.

Criptarea după cifrul Cezar poate fi reprezentată folosind aritmetică modulară prin transformarea literelor în numere conform schemei A = 0, Ă = 1,..., Z = 30. Astfel, alfabetul devine o secvență de 31 de numere, iar criptarea unei litere cu poziția din alfabet {\displaystyle x} printr-o deplasare spre dreapta cu *n* poziții poate fi descrisă matematic ca

{\displaystyle E\_{n}(x)=(x+n)\mod {31}.}

Decriptarea este făcută în mod similar:

{\displaystyle D\_{n}(x)=(x-n)\mod {31}.}

(Există mai multe definiții pentru operația modulo. În operația de mai sus, rezultatul se află în intervalul 0...30. Dacă *x+n* sau *x-n* nu se află în intervalul 0...30, atunci prin operația modulo se scad sau se adună 31 de atâtea ori până când condiția este îndeplinită).

***#COD***

alphabet = "abcdefghijklmnopqrstuvwxyz " #declaram alfabetul

letter\_to\_index = dict(zip(alphabet, range(len(alphabet))))

#facem conversia din litere in numarul de ordine din alfabet

index\_to\_letter = dict(zip(range(len(alphabet)), alphabet))

#facem conversia din numarul de ordine din alfabet in litere

Criptarea

def encrypt(message, shift=3): # functia pentru criptare cu parametrii mesaj si keye

cipher = "" # variabila pentru a salva mesajul criptat

for letter in message:#cream un ciclu pentru fiecare litera din mesajul clar

number = (letter\_to\_index[letter] + shift) % len(letter\_to\_index) #adunam numarul de ordine al literei cu keya apoi facem mod 26 (lungimea alfabetului)

letter = index\_to\_letter[number] #facem conversia numarului in litera corespunzatoare din alfabet

cipher += letter #o scrim in variabila noastra

return cipher #returnam mesajul criptat

Decriptarea

def decrypt(cipher, shift=3):# functia pentru decriptare cu parametrii mesaj si keye

decrypted = "" # variabila pentru a salva mesajul decriptat

for letter in cipher: #cream un ciclu pentru fiecare litera din mesajul criptat

number = (letter\_to\_index[letter] - shift) % len(letter\_to\_index) #scadem din numarul de ordine al literei keya apoi facem mod 26 (lungimea alfabetului)

letter = index\_to\_letter[number] #facem conversia numarului in litera corespunzatoare din alfabet

decrypted += letter #o scrim in variabila noastra

return decrypted #returnam mesajul clar

Corpul

def main():

message = input("Introduceti un mesaj:\n") # citim de la tastatura mesajul clar

key = int(input("Introduceti key-a:\n")) # citim de la tastatura key-a

cipher = encrypt(message, shift=key) # salvam in variabila cipher raspunsul returnat de functia de criptare (mesajul criptat)

decrypted = decrypt(cipher, shift=key) # salvam in variabila decrypted raspunsul returnat de functia de decriptare (mesajul decriptat)

print('Mesaj Original: ' + message) # Afisam mesajul clar

print('Mesaj Criptat: ' + cipher) #afisam mesajul criptat

print('Mesaj Decriptat: ' + decrypted) #afisam mesajul decriptat

main()

***#Screenshots***



