

SAE Crypto

Droucheau Quentin , Gnaneswaran Roshan

Message à décoder :

```
BDQE PG OTQYUZ EQ OMOTQ GZ FDQEAD
MOODAOTQ M GZ MDNDQ FAGF DQOAGHQDF P'AD
ZQ ZQSXUSQ BME XM VQGZQ BAGOQ RQGUXXG
SDMZP QEF EAZ EQODQF YMXSDQ EM FMUXXQ YQZGQ
DAZPQE QF OAXADQQE EAZF XQE NMUQE CG'UX BADFQ
MZUEQQE QF EGODQQE, XQGDE EMHQGDE EAZF RADFQE.
YMUE MFFQZFUAZ M ZQ BME XQE ODACGQD,
YQYQ EU XM RMUY FUDMUXXQ FQE QZFDMUXXQE,
QZ MGOGZ OME FG ZQ PAUE EG00AYNQD
```

Tout d'abord nous avons essayé en premier le chiffrement de César. En premier temps nous avons créés un dictionnaire de l'alphabet avec en clef la lettre et valeur un indice commençant à partir de 0.

```
alphabet = dict()
i = 0
for lettre in "ABCDEFGHIJKLMNOPQRSTUVWXYZ":
    alphabet[lettre] = i
    i += 1
```

Par la suite nous avons donc utilisé le dictionnaire pour créer la méthode de déchiffrement de César.

```
def dechiffrement_cesar(cle, texte):
    """
    Déchiffre un chiffrement de César avec une clé donnée.

    Args:
        cle (int): La clé utilisée pour déchiffrer le texte.
        texte (str): Le texte à déchiffrer.

    Returns:
        str: Le texte déchiffré.
    """
    global alphabet
    texte_decode = ""
    for lettre in texte:
        if lettre in alphabet.keys():
            nb = (alphabet.get(lettre) + cle) % 26

            for k in alphabet.keys():
                if alphabet[k] == nb:
                    texte_decode += k
```

```

        else:
            texte_decode += lettre
    return texte_decode

```

Afin de déterminer quel était le décalage nous avons fait une fonction permettant de tester les 26 décalages possibles.

```

def test_tous_les_decalage(texte):
    """
    Cette fonction prend en entrée un texte et affiche le texte chiffré
    avec tous les décalages possibles.

    Args:
        texte (str): Le texte à chiffrer.
    """
    for decalage in range(26):
        print(f"Texte chiffré avec un décalage de {decalage} :
        {dechiffrement_cesar(decalage, texte)}")

```

Pour plus de visibilité dans le terminal nous avons testé sur un mini texte et c'est ainsi qu'on a vu que le décalage était de 14.

```

texte = "BDQE PG OTQYUZ EQ OMOTQ GZ FDQED MOODAOTQ M GZ MDNDQ FAGF
DQOAGHQDF P AD ZQ ZQXSUSQ BME XM VQGZQ BAGOQ RQGUXXG SDMZP QEF EAZ EQODQF
YMXSDQ EM FMUXXQ YQZGQ DAZPQE QF OAXADQQE EAZF XQE NMUQE CG'UX BADFQ
MZUEQQE QF EGODQQE, XQGDE EMHQGDE EAZF RADFQE YMUE MFFQZFUAZ M ZQ BME XQE
ODACGQD, YQYQ EU XM RMUY FUDMUXXQ FQE QZFDMUXXQE, QZ MGOGZ OME FG ZQ PAUE
EGOOAYNQD"
mini_texte = "BDQE PG OTQYUZ EQ OMOTQ GZ FDQED MOODAOTQ M"
test_tous_les_decalage(mini_texte)

print(dechiffrement_cesar(14, texte))

```

Afin d'exploiter le fichier txt donnée dans le sujet nous avons une fonction pour passer du fichier chiffré en un fichier déchiffré.

```

def dechiffrer_fichier_cesar(fichier, cle, fichier_sortie):
    """
    Cette fonction prend en entrée un fichier et affiche le texte
    déchiffré avec tous les décalages possibles.

    Args:
        fichier (str): Le nom du fichier à déchiffrer.
        cle (int): La clé utilisée pour déchiffrer le texte.
        fichier_sortie (str): Le nom du fichier de sortie.
    """
    with open(fichier, "r") as f:

```

```
    texte = f.read()
    texte_decode = dechiffrement_cesar(cle, texte)
    with open(fichier_sortie, "w") as f:
        f.write(texte_decode)

dechiffrer_fichier_cesar("indice1_chiffre.txt", 14,
                        "indice1_dechiffre.txt")
```

Message trouvé :

PRES DU CHEMIN SE CACHE UN TRESOR
ACCROCHE A UN ARBRE TOUT RECOUVERT D'OR
NE NEGLIGE PAS LA JEUNE POUCE FEUILLU
GRAND EST SON SECRET MALGRE SA TAILLE MENUE
RONDES ET COLOREES SONT LES BAIES QU'IL PORTE
ANISEES ET SUCREES, LEURS SAVEURS SONT FORTES.
MAIS ATTENTION A NE PAS LES CROQUER,
MEME SI LA FAIM TIRAILLE TES ENTRAILLES,
EN AUCUN CAS TU NE DOIS SUCCOMBER

Le code caché dans le message est le suivant en prennant les premières lettres de chaque ligne :

PANGRAMME