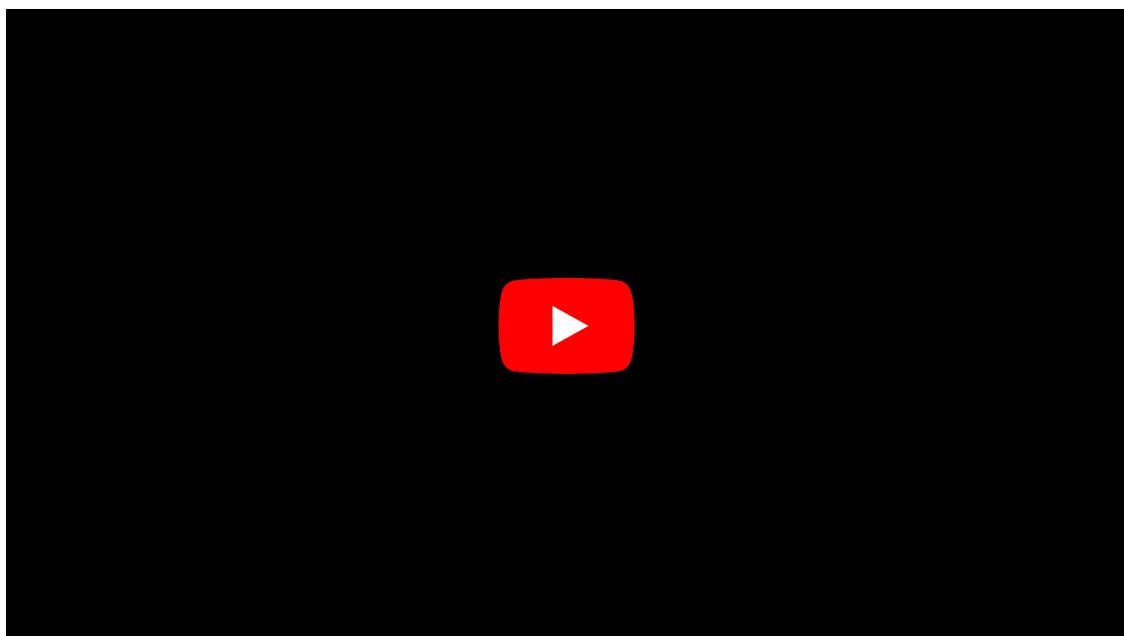


Découverte de la cryptographie

la cryptographie est une des applications majeures de l'informatique. Le but de projet est de réaliser un programme permettant de chiffrer et de déchiffrer un texte à l'aide du code de César, l'une des plus anciens (et des plus simples) méthodes de cryptage. Le code de César peut être cassé par analyse fréquentielle, on programmera donc aussi le décryptage du code de César en utilisant cette technique. Une autre partie du projet est consacrée au chiffrement de Vigenère.

1. Etape 1 : le code de César

Comprendre le code de César et la méthode pour le déchiffrer. On pourra faire ses propres recherches ou consulter la vidéo suivante (en anglais, mais les sous-titres français sont disponibles):



Sans l'aide d'un ordinateur, "à la main" :

- Chiffrer le mot "EXPERT" en décalant les lettres de 5 emplacements.
- Déchiffrer le mot "DOBBSLVO" en sachant que la clé est 10.

Préparer une explication orale de la méthode de chiffrement ainsi que la technique de déchiffrement, inclure des exemples.

2. Etape 2 : Chiffrer ou déchiffrer avec la clé

Réaliser un programme permettant à un utilisateur de chiffrer ou de déchiffrer un texte avec la clé de son choix. Pour simplifier, on suppose que le texte est constitué uniquement de *lettres majuscules non accentuées*, de ponctuations et d'espaces. On écrira une fonction `chiffre_texte` qui prend en argument une chaîne de caractère `texte` et une clé

`cle` (entier compris entre 1 et 25) et qui renvoie `texte` chiffré avec le code de César de clé `cle` (on laisse intacte les espaces et caractères de ponctuation, on ne déchiffre que les lettres majuscules). Par exemple :

Console Python

```
>>> chiffre_texte("NSI", 14)
'BGW'
>>> chiffre_texte("SUPER", 5)
'XZUJW'
>>> chiffre_texte("C'EST GENIAL", 11)
"N'PDE RPYTLW"
```

Aide

- Penser à utiliser les fonctions `ord` et `chr` de Python
- On pourra commencer par écrire une fonction `chiffre_caractere` qui prend en argument un caractère `caractere` et une clé `cle` et renvoie `caractere` décalé de `cle` emplacements.

3. Etape 3 : Analyse fréquentielle

Faire un programme qui décrypte automatiquement un texte crypté par la méthode de César grâce à une analyse fréquentielle. Tester votre programme sur l'exemple suivant :

Texte

```
PFOJC JCIG OJSN FSIGGW O RSQCRSF QS ASGGOUS
Q SGH HFSG PWSB AOWG WZ FSGHS SBQCFS PSOIQCID O TOWFS
```

Aide

On pourra commencer par écrire une fonction `plus_frequent` qui prend en argument un texte et renvoie la lettre qui apparaît le plus souvent dans ce texte. Par exemple :

Console Python

```
>>> plus_frequent("UN EXEMPLE DE TEXTE")
'E'
```

4. Etape 4 : dictionnaire et force brute

Tester votre programme sur le déchiffrement automatique du texte suivant :

Texte

RU YXDBBJ DW YAXOXWM BXDYRA, B' JBBRC MJWB BXW URC, B' JYYDHJWC BDA BXW YXUXLQXW. RU YARC DW AXVJW, RU U'XDEARC, RU UDC; VJRB RU W'H BJRBRBBJRC ZD'DW RVKAXPURX LXWODB, RU KDCJRC À CXDC RWBCJWC BDA DW VXC MXWC RU RPWXAJRC UJ BRPWRORLJCRXW.

Ce texte est extrait du livre *la disparition* (G. Perec), faire des recherches sur ce livre et expliquer sa particularité. Expliquer alors pourquoi le programme de déchiffrement par analyse fréquentielle ne fonctionne pas.

Une autre méthode de décryptage consiste à utiliser une **attaque par force brute** c'est à dire qu'on teste toutes les clés possibles et on vérifie que les mots obtenus après déchiffrement sont des mots du dictionnaire. Mettre en oeuvre en Python cet algorithme de décryptage et l'utiliser pour le texte donné en exemple ci-dessus.

Aide

On pourra télécharger ci-dessous un dictionnaire des mots de la langue française :

[Dictionnaire !\[\]\(cbe2492b119e39e02a1dab2af4a4b296_img.jpg\)](#)

5. Etape 5 : Codage de Vigenère

Faire des recherches sur le codage de Vigenère, expliquer son fonctionnement, détailler un exemple de codage avec cette méthode. Ce code résiste-t-il à une approche par analyse fréquentielle ? Justifier.

Proposer un programme Python permettant de coder et de décoder un texte avec la méthode de chiffrement de Vigenère.

Bibliographie :

- Ce cours s'inspire largement de celui de Fabrice Nativel.