« Mini projet »

Le Chiffre de César

1 Le chiffrement par décalage

Le chiffre de César est la méthode de cryptographie la plus ancienne communément admise par l'histoire. Il consiste en une substitution **mono-alphabétique**.

1.1 À propos de Jules César

Jules César était un général, homme politique et écrivain romain, né à Rome le 12 juillet ou le 13 juillet 100 av. J.C. et mort le 15 mars 44 av. J.C. Il aurait été assassiné par une conspiration, son propre fils Brutus lui portant le coup de grâce.
César s'est illustré lors de la guerre des Gaules, ce qui a donné des siècles plus tard son personnage dans la bande dessinée Astérix le Gaulois. Il utilisait une méthode de chiffrement qui porte aujourd'hui son nom.

1.2 Principe

En *cryptographie*, le chiffrement par décalage, aussi connu comme le chiffre de César, est une méthode de chiffrement très simple utilisée par Jules César dans ses correspondances secrètes (ce qui explique le nom « chiffre de César »).
Le texte chiffré s'obtient en remplaçant chaque lettre du texte clair original par une lettre à distance fixe, toujours du même côté, dans l'ordre de l'alphabet. Pour les dernières lettres (dans le cas d'un décalage à droite), on reprend au début.

Par exemple avec un décalage de 3 vers la droite, A devient D, B devient E, ...

$$A \longmapsto D \quad B \longmapsto E \quad C \longmapsto F \quad \dots \quad W \longmapsto Z \quad Y \longmapsto A \quad Z \longmapsto B$$

Il s'agit d'une permutation **circulaire** de l'alphabet.

La longueur du décalage, 3 dans l'exemple évoqué, constitue la clé de chiffrement qu'il suffit de transmettre au destinataire, s'il sait déjà qu'il s'agit d'un chiffrement de César, pour que celui-ci puisse déchiffrer le message.

1.3 Sécurité

Niveau sécurité, le chiffre de César n'est pas fiable du tout, et ce pour deux raisons :

- Il n'existe que 26 façons différentes de crypter un message : puisqu'on ne dispose que de 26 lettres, il n'y a que 26 décalages possibles. Dès lors, des attaques exhaustives (tester tous les décalages un à un) ne demanderaient que très peu de temps.
- Le chiffre de César est très vulnérable à l'analyse des fréquences.

Toutefois sa simplicité en fait un très bon exercice pédagogique, de plus celui-ci est régulièrement utilisé sur les forums de discussions en ligne pour brouiller tout ou partie d'un texte (comme la chute d'une blague, ou un spoiler), mais pas comme méthode de chiffrement en tant que telle.

1.4 Exemple

Le chiffrement peut être représenté par la superposition de deux alphabets, l'alphabet clair présenté dans l'ordre normal et l'alphabet chiffré décalé, à gauche ou à droite, du nombre de lettres voulu. Nous avons ci-dessous l'exemple d'un encodage de 3 lettres vers la droite. Le paramètre de décalage (ici 3) est la clé de chiffrement :

```
clair   : ABCDEFGHIJKLMNOPQRSTUVWXYZ
chiffré : DEFGHIJKLMNOPQRSTUVWXYZABC
```

Pour encoder un message, il suffit de regarder chaque lettre du message clair, et d'écrire la lettre encodée correspondante. Pour déchiffrer, on fait tout simplement l'inverse.

```
original : LE CHIFFRE DE CESAR
encodé   : OH FKLIIUH GH FHVDU
```

2 Papier, crayon, action ...

Le but de ce mini projet est de développer une interface avec Python et la librairie Tkinter pour obtenir un logiciel permettant de chiffrer et de déchiffrer un texte par décalage. Nous fixons arbitrairement cette valeur de décalage à 8. Toutefois le client souhaite pouvoir choisir au moment du chiffrement la clé (nombre de décalage).

De plus une option de lire un fichier en entrée pour effectuer le décalage doit être proposée.

2.1 Implémentation

Avertissement !
Avez-vous bien réfléchi à votre algorithme ? Si oui vous pouvez passer à l'implémentation sinon papier, crayon et action !

Voici quelques conseils qui pourraient vous être utile :

À propos du « `for` » en Python
La boucle `for` de Python fonctionne comme un itérateur sur un élément itérable (liste, tuple, dictionnaire, ...).
Même les chaînes de caractères sont des objets itérables en effet elles contiennent une séquence de caractères.

Quelques instructions en Python 3

```
1 ord('s') # get ascii value of given char
2 chr(126) # convert given ascii value to char (a string of length 1) here
  ↪ 126 stand for ~
3 ';'.join(['C', 'e', 's', 'a', 'r', ]) # return a string equal to
  ↪ C;e;s;a;r
```

2.2 Consigne pour le rendu

Le rendu de votre devoir se fera via un dépôt git (public) qui contiendra les sources de votre projet, un fichier README indiquant les noms des étudiants du groupe et les fonctionnalités implémentés au sein du projet.

Votre rendu est attendu pour le 8 juillet 2021 23h59. Vous transmettrez le lien vers votre dépôt GIT à l'adresse mail suivante : delahayeyourself@gmail.com

`Alea jacta est`