PROJET 1ARI

Soufian Oualla
Lounes Behloul
Cyprien Taïb

Quelques questions générales

1. Comment qualifier cet algorithme ?

C'est un algorithme à complexité exponentielle. Ce qui peut rendre l'algorithme très lent selon la taille des données à traiter. La création de clé est de complexité quadratique et l'encodage et le chiffrement le sont aussi. C'est donc un algorithme lent.

2. Quel est le nombre de clés de cet algorithme ? On distinguera l'entier et la grille.

Dans l'alphabet français, il existe 25 lettres sans le 'w'. Il est possible d'effectuer 25 factoriels de substitution. En effet, dans cet algorithme, les lettres étant systématiquement remplacées par les mêmes bigrammes, cela correspond donc au nombre de substitutions pour chacune des 25 lettres.

3. Que penser de la sécurité de cet algorithme ? On précisera ses qualités et défauts.

C'est un algorithme qui sécurise plutôt bien les données, de plus il est relativement simple à réaliser et à mettre en pratique. C'est un chiffrement symétrique, c'est-à-dire que la clé utilisée lors du chiffrement est aussi la même que celle utilisée lors du déchiffrement, ce qui le rend donc moins complexe et beaucoup plus facile à mettre en place. Néanmoins le problème avec ce type de chiffrement est la difficulté à communiquer la clé de manière sûre. Donc si une personne intercepte le message elle pourra le déchiffrer facilement voire même falsifier le contenu du message. De plus, il s'agit d'un chiffre polygrammique qui remplace une lettre par le même groupe de lettres, il existe donc une possibilité de décryptage

4. Proposer une méthode de décryptement de cet algorithme.

Chaque lettre étant associée à un bigramme, on peut tout d'abord regrouper le message par bigramme. Dans ce chiffrement les lettres étant remplacées par les mêmes bigrammes, on peut donc analyser la fréquence de parution des bigrammes afin de les comparer à la fréquence de parution des lettres de la langue en question. Étant donné que l'on utilise une grille de 25 lettres, on obtient donc 25 bigrammes différents. On réalise donc une substitution grâce à l'analyse des fréquences afin de tenter de décrypter le message.

PROJET 1ARI

Soufian Oualla
Lounes Behloul
Cyprien Taïb

Nos démarches

1. **Analyse du chiffre de Collon**

   Avant toute chose, nous avons commencé par analyser le fonctionnement du processus de chiffrement et de déchiffrement de Collon.

   Son principe de base est le suivant, La clé de ce chiffre est constituée d'un entier naturel, par exemple sept, et d'un ensemble de lettres complété par le reste de l'alphabet sans répétition (sans le 'w') dans un carré de cinq cases. Pour faire la grille, on constitue un ensemble de cinq sous listes de longueur cinq lettres.

2. **Lecture, déchiffrement et chiffrement d'une lettre**
   On a remarqué que l'on devait parcourir une grille afin d'extraire des coordonnées basées sur la dernière ligne et la première colonne
   Comme la convention veut que les carrés soit des listes de listes, pour chiffrer et déchiffrer une lettre il suffisait de prendre la lettre à l'indice de la lettre dans la dernière lettre et l'autre la première lettre de la sous-liste où a été trouvé la lettre.

3. **Chiffrer un texte et le déchiffrer**
   Maintenant que l'on a l'algorithme de chiffrement et de déchiffrement d'une lettre, on parcourt un texte prédécoupé afin de le chiffrer ou de le déchiffrer.
   .
4. **Création du carré et découpage du texte**

   Pour découper un texte on enlève la lettre w et les espaces, on demande à l'utilisateur l'entier n et on va insérer un espace tous les n caractère.

   Pour la création du carré on va juste remplacer tous les 'w' par des 'v' et on va ajouter les lettres restantes de l'alphabet puis les mettre en sous-liste de cinq.

5. **Réalisation de l'interface graphique**

PROJET 1ARI

Soufian Oualla
Lounes Behloul
Cyprien Taïb

L'interface a été réalisée en Tkinter. Le format de base a été de répartir les différents styles de widgets en plusieurs frames, le text_entry par exemple est à l'intérieur d'une frame, les données qui constituent la clé sont dans une autre frame, etc. La première étape a été de formater le tkinter de façon à suivre certaines conventions du sujet mais aussi pour avoir un visuel acceptable pour utiliser les fonctionnalités, ensuite il a fallu lier les différentes parties du code à l'interface.

Le premier élément de liaison a été de pouvoir manipuler le texte que l'utilisateur a saisi pour pouvoir effectuer les différentes actions de chiffrement et de déchiffrement en tenant compte de la clé, on a pu réaliser cela assez simplement avec les fonctions internes de tkinter, le .get() pour récupérer les informations qui ont ensuite été manipulés avec une classe créée; textCypher une classe pour lier le code au tkinter afin d'éviter certains problèmes, et faciliter l'utilisation de cypher et uncypher. On a ensuite ajouté des binds pour contrôler de manière plus facile les interactions avec l'interface.

6. Fonctionnalités supplémentaires

Une fois notre interface totalement fonctionnelle, nous avons pu améliorer notre code en y ajoutant des fonctionnalités supplémentaires. Notamment la possibilité de personnaliser l'interface graphique en pouvant changer la police d'écriture ou encore en pouvant augmenter le type de police. D'ailleurs, il est aussi possible d'enregistrer et de charger un ancien chiffrement.