



Projet Codez pour décoder

Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : Codez pour décoder

L'on choisit un mot clé auquel on se référera (en l'occurrence ROUGE) pour trouver les lignes à regarder dans le tableau afin de chiffrer le message et l'on regarde pour chaque lettre du message sur la première ligne, la colonne qui lui correspond. Si le mot clé est plus court que le message, la lettre suivante du message sera codée avec la première lettre du mot clé, et cela jusqu'à la fin du message.

- Un autre type de cryptage assez simple mais très efficace est le chiffrement aléatoire. Cette méthode consiste à insérer entre chacune des lettres un nombre de lettres défini. Par exemple, si on choisit 2 lettres pour coder le mot ROUGE, le programme va insérer entre chaque lettre deux lettres complètement aléatoires qui n'ont rien à voir avec le message ce qui va rendre le message totalement illisible pour quelqu'un qui ne sait pas quelles sont les lettres inutiles. Ce chiffrement a cependant un défaut : en s'armant de courage, on peut essayer toutes les clés jusqu'à trouver la bonne.

- Enfin nous avons intégré le « Leet speak », un langage que nous avons utilisé pour l'interface générale pour avoir un rendu plutôt esthétique et original à regarder : un écran interactif. Cette façon de coder un message consiste à remplacer chaque lettre de l'alphabet par une équivalence physique grâce à des caractères comme les tirets, les parenthèses, etc.

• C'est grâce à ces différentes manières de chiffrer un message que « codez pour décoder » a été créé. Désormais vous pouvez être sûr qu'un message que vous vous faites passer en classe restera incompréhensible pour votre professeur !!

> ORGANISATION DU TRAVAIL :

- *Présentation de l'équipe (prénom de chaque membre et rôle dans le projet)*
- *Répartition des tâches*
- *Organisation du travail (répartition par petits groupes, fréquence de réunions, travail en dehors de l'établissement scolaire, outils/logiciels utilisés pour la communication et le partage du code, etc.)*

CASSOU Antoine : création des boutons, organisation tkinter, monteur vidéo ;

SOUCHON-GARRIGUE Manon : organisation tkinter, gestionnaire de l'interface utilisateur, esthétique et signature graphique ;

VIROULAUD Abel : création et écriture des fonctions, compte rendu écrit .

Notre travail est assez partagé avec comme répartition, l'un qui s'occupe des fonctions de cryptage et de décryptage pendant qu'un autre s'occupe des boutons et que le dernier met en place et améliore l'interface graphique en parallèle de mises en commun régulières.

Avec des mises en commun hebdomadaires et un travail intense en dehors de l'établissement (en effet tout ce projet s'est construit entièrement en dehors du temps scolaire) nous sommes finalement, après un grand nombre d'heures d'investissement personnel, arrivés à nos fins.

Nous n'avons pas créé de groupe de travail chacun avançait à son rythme chez lui puis venait à la réunion, cependant nous échangeons régulièrement des messages via instagram pour trouver des idées ou pour se mettre d'accord rapidement en dehors des réunions hebdomadaires. Nous avons également échangé par email pour se transférer facilement les documents et des clés USB pour le code.

LES ÉTAPES DU PROJET :

- *Présenter les différentes étapes du projet (de l'idée jusqu'à la finalisation du projet)*

Films, livres et podcasts sont à l'origine de ce projet. Nous avons commencé par écrire le code Vigenère ainsi que le code César en classe... C'est notre point de départ, puis, l'idée du concours et là se pose le problème du sujet. Tous trois très intéressés par la partie du cours de NSI qui concernait le chiffrement d'informations, nous sommes partis de là.

Premièrement, il a fallu trouver différentes manières de coder un message. Nous en avons trouvé quatre, puis Antoine en a déniché une cinquième dans son livre The Code Book de Simon Singh.

Après avoir réussi à faire marcher les programmes, nous nous sommes lancés dans l'interface utilisateur. Nous nous sommes servis du module Tkinter . Très rapidement la première version sans fond d'écran, sans placement et sans style était créée. Mais la base était là, la centralisation des systèmes de cryptage était née.

Fonctionnel mais... pas très ergonomique. Au fur et à mesure de l'avancement du projet nous avons rajoutés des boutons, un fond, des indications pour une utilisation guidée et simple.

Puis continuant dans cette lancée nous avons décidé que lorsque la souris passe sur un bouton, celui-ci s'afficherait en leet speak.

Vient ensuite la partie organisation du code afin qu'il soit lisible et propre.

Enfin nous nous sommes retrouvés un mercredi après midi pour faire la vidéo et se mettre au clair sur les choix techniques, la présentation, les idées d'ouverture et d'amélioration.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

- *Avancement du projet (ce qui est terminé, en cours de réalisation, reste à faire)*
- *Approches mises en œuvre pour vérifier l'absence de bugs et s'assurer de la facilité d'utilisation du projet*
- *Difficultés rencontrées et solutions apportées*

Le projet en lui même est terminé, nous pouvons toujours trouver des améliorations mais nous sommes arrivés au résultat que nous attendions. L'idée en elle-même reste en réalisation car nous pourrions ajouter d'autres systèmes de cryptage et de décryptage. Nous pourrions améliorer l'interface utilisateur avec un page d'accueil qui présenterait l'application.

Pour éviter les bugs ou autre, nous avons tout d'abord testé les programmes de décryptage un à un en testant les exceptions (accent, espace, etc.) puis nous avons vérifié que tous les boutons marchaient et que l'interface utilisateur était correctement organisée. Pour finir nous avons réuni les deux en utilisant des messages préalablement traduits et en les faisant coder ou décoder par notre programme en utilisant l'interface utilisateur pour tester en parallèle le cryptage et l'affichage.

Nous avons rencontré deux grandes difficultés. La principale difficulté rencontrée a été de s'adapter à un nouveau module, Tkinter. En effet nous ne l'avions jamais utilisé et il était à la base de notre interface graphique. C'est de là qu'est survenue la deuxième difficulté rencontrée : le placement des éléments sur l'interface graphique. Écrire les codes pour crypter ou décrypter est certes compliqué et parfois on fait des pauses pour réfléchir et trouver une solution à un certain problème mais centrer les boutons et gérer les effets graphiques a été long et fastidieux comme nous ne maîtrisons pas très bien le module.

Nous avons donc regardé des tutos et demandé conseils et astuces à notre professeur de NSI. Pour le placement des boutons nous avons tâtonné au fur et à mesure en utilisant `.place()`. Nous avons les coordonnées d'un des point de l'angle du bouton et après plusieurs essais pour chacun des boutons, nous leur avons trouvé la bonne place. Ensuite nous avons découvert les fonctionnalités bien plus simples de `.pack()` ce qui nous a facilité la tâche.

> OUVERTURE :

- *Idées d'améliorations (nouvelles fonctionnalités)*
- *Stratégie de diffusion pour toucher un large public (faites preuve d'originalité !)*
- *Analyse critique du résultat (si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?)*

Nous avons plusieurs idées d'améliorations. Bien évidemment d'autres types de cryptage (ou même d'autres niveaux de leet speak) ainsi qu'une fonction « partager » afin de rendre le projet utilisable de manière très rapide et efficace. Une option pour coller le message chiffré ou déchiffré est aussi envisageable.

Notre principal axe de recherche reste néanmoins d'insérer la fonction « Markov Chain Monte Carlo » afin de pouvoir déchiffrer n'importe quel autre message codé.

Afin de toucher le public le plus large possible nous envisageons une diffusion sur la plupart des réseaux sociaux avec un publicité qui mettrait en avant l'importance du cryptage de nos jours suivis de notre projet qui permet une communication simple et rapide. Cette amélioration prendrait en compte le fait de pouvoir enregistrer nos images en ligne et ainsi pouvoir exécuter le programme en étant redirigé vers un exécuteur python en ligne. Cependant un problème se pose car tout le monde y aurait accès, ainsi il faudrait utiliser des doubles cryptages ou de nouvelles manières de chiffrer... Autant d'opportunité que possible pour faire découvrir la NSI à de nouvelles communautés.

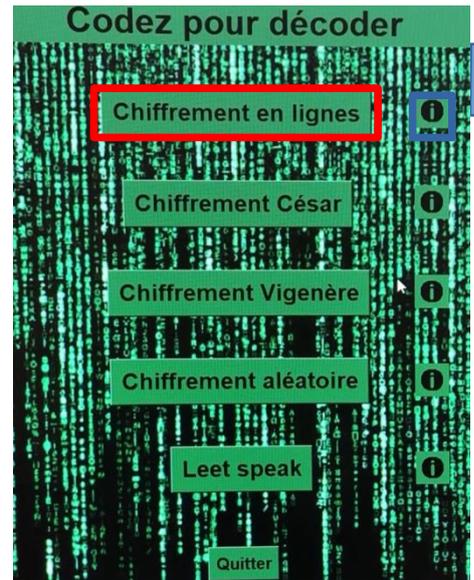
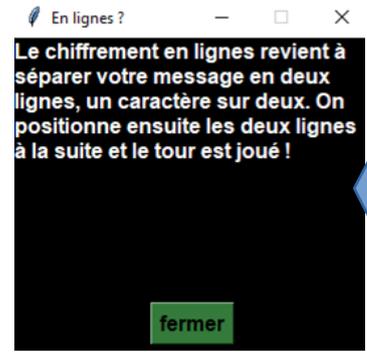
Je pense que si nous devons le refaire, le changement principale serait de s'y prendre un peu plus tôt. En effet nous ne tirons que du positif de cette aventure, des souvenirs, des habitudes de programmation, la découverte d'un nouveau langage, une progression assez significative entre le début où coder un bouton nous paraissait démesuré et la fin où nous avons rajouté une multitude de détails et de nouvelles fonctions très rapidement. Cependant, avec plus de temps nous aurions sûrement essayé de faire en sorte d'inverser les couleurs des boutons info lorsqu'on clique dessus, ou encore d'afficher le bouton « Fermer » des fenêtres d'information nous renseignant sur la particularité du type de chiffrement en leet speak, comme pour les autres boutons.

Pour diffuser notre projet, une mise à disposition du code dans les ordinateurs de notre établissement serait judicieuse. Ainsi le but de cette interface serait atteint : que des personnes puissent s'amuser en chiffrant et déchiffrant des messages ! Ainsi nous travaillerions le code tous les mois et améliorerions d'une ou plusieurs méthodes de chiffrement chaque fois pour créer de la nouveauté et que les élèves ne se lassent pas !

DOCUMENTATION

- *Spécifications fonctionnelles (guide d'utilisation, déroulé des étapes d'exécution, description des fonctionnalités et des paramètres)*
- *Spécifications techniques (architecture, langages et bibliothèques utilisés, matériel, choix techniques, format de stockage des données, etc)*
- *Illustrations, captures d'écran, etc*

Notre page s'affiche en exécutant le fichier python grâce au module Tkinter, en ayant deux photos (png et jpeg) et un gif dans le même dossier. Notre interface se compose d'une « frame » principale et de plusieurs sous-frames qui apparaissent si l'on clique sur le bouton associé. Dans chaque sous-frame se trouve un chiffrement sur lequel on peut trouver quelques informations en cliquant sur le « i » à côté de son nom sur la frame principale. Les sous-frames se composent donc d'un espace pour entrer son message (ainsi que d'une clé si besoin pour le chiffrer). On clique sur le bouton « chiffrer » et la fonction du chiffrement choisi s'exécute puis le message ressort chiffré de la manière souhaitée dans l'espace en dessous. C'est le même principe pour le déchiffrement. On peut également fermer la fenêtre grâce au bouton « quitter » en bas de la frame principale.



- *  Détails sur le chiffrement
-  Choisir le chiffrement



- ↑ Insérez la clé du message si le chiffrement en requiert une
- ← Insérez un message
- ← curseur pour faire défiler le message
- ← sortie message chiffré

Activer la fonction pour chiffrer le message

Nous avons intégré deux easter eggs dans notre code. Ainsi lorsqu'on rentre 'r' ou 'R' dans le chiffrement d'un message, l'utilisateur est renvoyé vers cette vidéo youtube : Rick Astley - Never Gonna Give You Up (Official Music Video)

(<https://www.youtube.com/watch?v=dQw4w9WgXcQ>) qui est un meme très connu et utilisé, qui nous tenait à cœur d'inclure dans le projet. En outre un bouton noir est caché dans le fond d'écran. Lorsqu'il est cliqué, celui-ci renvoie vers le bêtisier de la vidéo que nous avons réalisée pour rendre le projet.



Chaque bouton s'affiche en leet speak lorsque la souris le survole, puis ses couleurs sont inversées s'il est cliqué (il affiche alors du texte vert sur un fond noir). Chaque bouton sauf les boutons info à droite des chiffrements sur la page principale ainsi que les boutons « fermer » dans les pages « info » ne respectent pas l'ensemble des deux conditions.



Nous avons choisi de mettre en icône de la page une tête dans laquelle sont inscrits des nombres à la manière du fond d'écran de notre page, en référence au film Matrix des frères Wachowski (cf : ci-contre). Cette icône représente la pensée et la réflexion (tête) mais également la programmation (chiffres type Matrix). C'est donc une référence directe à notre parcours dans les Trophées NSI et à notre manière de penser la spécialité NSI.

