

Cryptographie

Léa MENERET, Fathima SAHADATTALY, Ulrike KULZER
21 décembre 2017

Table des matières

1 Guide utilisateur	2
1.1 Introduction	2
1.1.1 En général	2
1.1.2 Méthodes de cryptage	2
1.2 Fonctionnement du programme	4
1.2.1 Accueil et choix de la langue	4
1.2.2 Crypter/Décrypter	5
1.2.3 Choix du principe	5
2 Interfaces utilisateurs	6
2.1 En français	7
3 Guide de maintenance	10
3.1 Fonctionnalité	10
3.1.1 En général	10
3.1.2 Différents états du programme	10
3.1.3 En détail et coupé en modules	11
3.1.4 Structure du programme	12
4 Annexe : Code du programme	16

1 Guide utilisateur

1.1 Introduction

1.1.1 En général

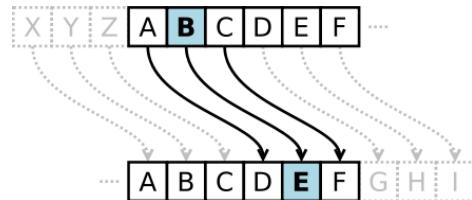
La cryptographie est une technique utilisée pour rendre incompréhensible à autrui un message entre un expéditeur et un destinataire. Ce procédé a notamment été utilisé en période de guerre pour permettre des attaques surprises. Le principe est le suivant : L'expéditeur à partir d'une clé crypte son message et l'envoie au destinataire. Celui-ci possède aussi la clé qui va lui permettre ainsi de décrypter le message.

1.1.2 Méthodes de cryptage

La cryptographie est utilisée depuis l'Antiquité mais certaines de ces méthodes les plus abouies datent du 20e siècle. Il existe différents principes de cryptage plus ou moins compliqués tels que

— le chiffre de César :

Ce procédé a été inventé lors de l'époque romaine par Jules César pour ses communications secrètes. En décalant l'alphabet par un entier donné chaque lettre est associée à une nouvelle lettre, ainsi on peut crypter le message initiale en remplaçant chaque lettre par la nouvelle lettre attribuée.



— le chiffre de Vigenère :

Il a été inventé au 16e siècle par Blaise de Vigenère et est basé sur le tableau à droite. Une clé (un mot) est répétée et mis sous le message et de cette manière on peut trouver les lettres correspondantes à partir du tableau.

Exemple :

		Lettre en clair																									
		26 lettres chiffrées																									
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z			
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z				
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z					
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z						
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z							
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z								
I	I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z								
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z										
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z											
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z												
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z													
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z														
O	O	P	Q	R	S	T	U	V	W	X	Y	Z															
P	P	Q	R	S	T	U	V	W	X	Y	Z																
Q	Q	R	S	T	U	V	W	X	Y	Z																	
R	R	S	T	U	V	W	X	Y	Z																		
S	S	T	U	V	W	X	Y	Z																			
T	T	U	V	W	X	Y	Z																				
U	U	V	W	X	Y	Z																					
V	V	W	X	Y	Z																						
W	W	X	Y	Z																							
X	X	Y	Z																								
Y	Y	Z																									
Z	Z																										

clé : musique

Texte : J'adore écouter la radio toute la journée.
Texte en claire et en dessous la clé répétée:

j'adore écouter la radio toute la journée.
M USIQU EMUSIQU EM USIQU EMUSI QU EMUSIQU
^ ^ ^

Colonne 0, ligne I : on obtient la lettre w.
Colonne D, ligne S : on obtient la lettre v.
Colonne A, ligne U : on obtient la lettre u.
Colonne J, ligne M : on obtient la lettre v.

— *celui de la machine Enigma :*

L'Enigma est une machine de cryptographie inventée par Arthur Scherbius en 1919. Elle a été utilisée durant la Seconde Guerre mondiale pour la communication secrète entre les différentes unités de l'armée allemande. La machine est constituée de cinq rotors dont un réflecteur, d'un clavier, d'un tableau de permutation et de lampes pour chaque lettre. Pour l'allumer il faut une batterie de 4,5 Volt. Le principe est simple : Lorsqu'on appuie sur une lettre du clavier, un courant électrique va être envoyé au tableau de permutation dans lequel la lettre entrée est échangée avec une autre lettre si elles sont connectées. Puis il passera la première fois par les quatre rotors : Dans chacun des trois rotors au milieu il y a un décalage des lettres qui s'opère. À la fin les lettres sont permutees encore une fois dans le réflecteur qui les renvoie par les rotors au tableau de permutation ce qui permettra à une lampe correspondant à une lettre de s'allumer. Ainsi pour chaque lettre on relève la lettre codée, on obtient alors notre message crypté.



Sites de référence :

- <https://fr.wikipedia.org/wiki/Cryptographie>
- <http://www.bibmath.net/crypto/>
- https://fr.wikipedia.org/wiki/Chiffrement_par_d%C3%A9calage
- https://fr.wikipedia.org/wiki/Chiffre_de_Vigen%C3%A8re
- [https://fr.wikipedia.org/wiki/Enigma_\(machine\)](https://fr.wikipedia.org/wiki/Enigma_(machine))
- photo d'Enigma : Von William Warby from London, England - Enigma, CC BY 2.0, <https://commons.wikimedia.org/w/index.php?curid=46848023>

1.2 Fonctionnement du programme

1.2.1 Accueil et choix de la langue

Pour exécuter notre programme, il est nécessaire de l'ouvrir avec le logiciel Pycharm, d'aller dans le fichier "main_module" et d'appuyer sur "run" - "run" - "run main_module". Les instructions du programme seront affichées dans la console qui s'ouvre en bas de l'interface comme montrés ci dessous :

```
Hello. Welcome to the cryptography program.  
Please select your language:  
'f' for Français, 'e' for English.
```

Fig. 1 – Écran d'accueil

Vous venez d'entrer dans le programme Cryptographie, vous devez entrer la lettre ' f ' pour avoir les instructions en français et ' e ' pour les avoir en anglais. Si la lettre entrée est ' e ', les prochaines instructions seront donc en anglais.

Note : Vous retrouverez les mêmes interfaces en français.

Une nouvelle fenêtre va ensuite apparaître :

```
What do you want to do - encrypt or decrypt a message?  
Enter 'c' for encrypting or 'd' for decrypting.  
Enter 's' to change settings (language)  
or 'q' to exit :(
```

Fig. 2 – Menu principal

Le programme demande maintenant ce que vous voulez faire, vous devez entrer ' c ' si vous voulez que le programme crypte votre texte ou ' d ' si vous voulez qu'il le décrypte. Si vous voulez changer de langue entrez ' s ' et si malheureusement vous voulez quitter le programme entrez ' q ' .

Si la lettre entrée est ' s ' , une nouvelle fenêtre permettant de changer la langue va alors apparaître.

```
Please select your language:  
'f' for Français, 'e' for English.
```

Fig. 3 – Paramètres

Vous devez entrer la lettre ' f ' pour avoir les instructions en français et ' e ' pour les avoir en anglais.

Si la lettre entrée est ' f ' les prochaines instructions seront donc en français.
Vous reviendrez sur les instructions précédentes cette fois-ci en français.

1.2.2 Crypter/Décrypter

```
Bonjour. Bienvenue sur le programme de cryptographie.  
Qu'est-ce que vous voulez faire - crypter ou décrypter un message?  
Insérez 'c' pour crypter ou 'd' pour décrypter.  
Insérez 's' pour changer les paramètres (langue)  
ou 'q' pour quitter le programme :(
```

Fig. 4 – Menu principal

Vous devez entrer ' c ' si vous voulez que le programme crypte votre texte ou ' d ' si vous voulez qu'il le décrypte.

Si vous voulez changer de langue entrez ' s ' et si malheureusement vous voulez quitter le programme entrez ' q ' .

Si la lettre entrée est ' c ' , le programme va donc être servi pour le cryptage d'un texte. Il existe trois méthodes par lesquelles le texte peut être crypté, nous allons donc détailler les différentes manières.

1.2.3 Choix du principe

- *César* :
- Vigenère
- Enigma

2 Interfaces utilisateurs

2.1 En français

```
Please select your language:  
'f' for Français, 'e' for English.
```

Fig. 5 – Paramètres

```
De quelle manière voulez vous crypter votre texte?  
Insérez 'c' pour le cryptage par le chiffre de César,  
insérez 'v' pour le cryptage par le chiffre de Vigenère  
ou insérez 'e' pour le cryptage par la machine Enigma.  
Insérez 'm' pour retourner au menu principal.
```

Fig. 6 – Cryptage

```
De quelle manière voulez vous décrypter votre texte?  
Insérez 'c' pour le décryptage par le chiffre de César,  
insérez 'v' pour le décryptage par le chiffre de Vigenère  
ou insérez 'e' pour le décryptage par la machine Enigma.  
Insérez 'm' pour retourner au menu principal.
```

Fig. 7 – Décryptage

```
Vous avez choisi <<César>>:  
Insérez votre clé (un nombre compris entre 1 et 25).  
Insérez 'm' pour retourner au menu principal  
ou 'r' pour retourner au dernier menu.
```

Fig. 8 – Le chiffre de César

```
Vous avez choisi <<Vigenère>>:  
Insérez votre clé (un mot).  
Insérez 'm' pour retourner au menu principal  
ou 'r' pour retourner au dernier menu.
```

Fig. 9 – Le chiffre de Vigenère

```
Vous avez choisi <<Enigma>>:  
Insérez votre clé (composée de trois lettres en majuscule).  
Insérez 'm' pour retourner au menu principal  
ou 'r' pour retourner au dernier menu.
```

Fig. 10 – Le cryptage par la machine Enigma

```
Insérez votre texte.  
Insérez 'm' pour retourner au menu principal  
ou 'r' pour retourner au dernier menu.
```

Fig. 11 – Entrée du texte

```
Voici votre texte crypté:  
....
```

Fig. 12 – Affichage du texte crypté

```
Voici votre texte décrypté:  
....
```

Fig. 13 – Affichage du texte décrypté

```
Merci d'avoir utilisé notre programme.  
Bonne journée, au revoir.
```

Fig. 14 – Sortie du programme

***** AIDE *****

Voici les explications pour les différents principes de (dé)cryptage.

- Le chiffre de César :

Ce procédé a été inventé lors de l'époque romaine par Jules César pour ses communications secrètes. En décalant l'alphabet par un entier donné chaque lettre est associée à une nouvelle lettre, ainsi on peut crypter le message initiale en remplaçant chaque lettre par la nouvelle lettre attribuée.

- Le chiffre de Vigenère :

Il a été inventé au 16e siècle par Blaise de Vigenère et est basé sur le tableau de Vigenère (tableau avec deux fois l'alphabet). Une clé (un mot) est répétée et mis sous le message et de cette manière on peut trouver les lettres correspondantes à partir du tableau.

- Le principe de la machine Enigma:

L'Enigma est une machine de cryptographie inventée par Arthur Scherbius en 1919. Elle a été utilisée durant la Seconde Guerre mondiale pour la communication secrète entre les différentes unités de l'armée allemande.

La machine est constituée de cinq rotors dont un réflecteur, d'un clavier, d'un tableau de permutation et de lampes pour chaque lettre. Pour l'allumer il faut une batterie de 4,5 Volt.

Le principe est simple : Lorsqu'on appuie sur une lettre du clavier, un courant électrique va être envoyé au tableau de permutation dans lequel la lettre entrée est échangée avec une autre lettre si elles sont connectées. Puis il passera la première fois par les quatre rotors : Dans chacun des trois rotors au milieu il y a un décalage des lettres qui s'opère. À la fin les lettres sont permutees encore une fois dans le réflecteur qui les renvoie par les rotors au tableau de permutation ce qui permettra à une lampe correspondant à une lettre de s'allumer. Ainsi pour chaque lettre on relève la lettre codée, on obtient alors notre message crypté.

Insérez 'm' pour retourner au menu principal.

Fig. 15 – Aide

3 Guide de maintenance

3.1 Fonctionnalité

3.1.1 En général

- Quand l'utilisateur lance le programme on lui demande de choisir la langue, soit français, soit anglais.
- Notre programme va proposer à l'utilisateur trois façons différentes de (dé)crypter un texte de complexité croissante et demander une clé.
- Les différentes manières sont le principe du chiffre de César, du chiffre de Vigenère et de la machine Enigma.

3.1.2 Différents états du programme

Les différents états du programme sont représentés dans le diagramme ci-dessous. Dès que le programme est lancé, il est généralement toujours dans un état en attendant des saisies de l'utilisateur. La seule exception est l'état en cryptant ou décryptant. Sur les flèches on peut voir ce qu'il faut faire pour accéder à un autre état, soit le précédent, soit le suivant. Avant que le programme est fermé, il affichera un petit message de remerciement.

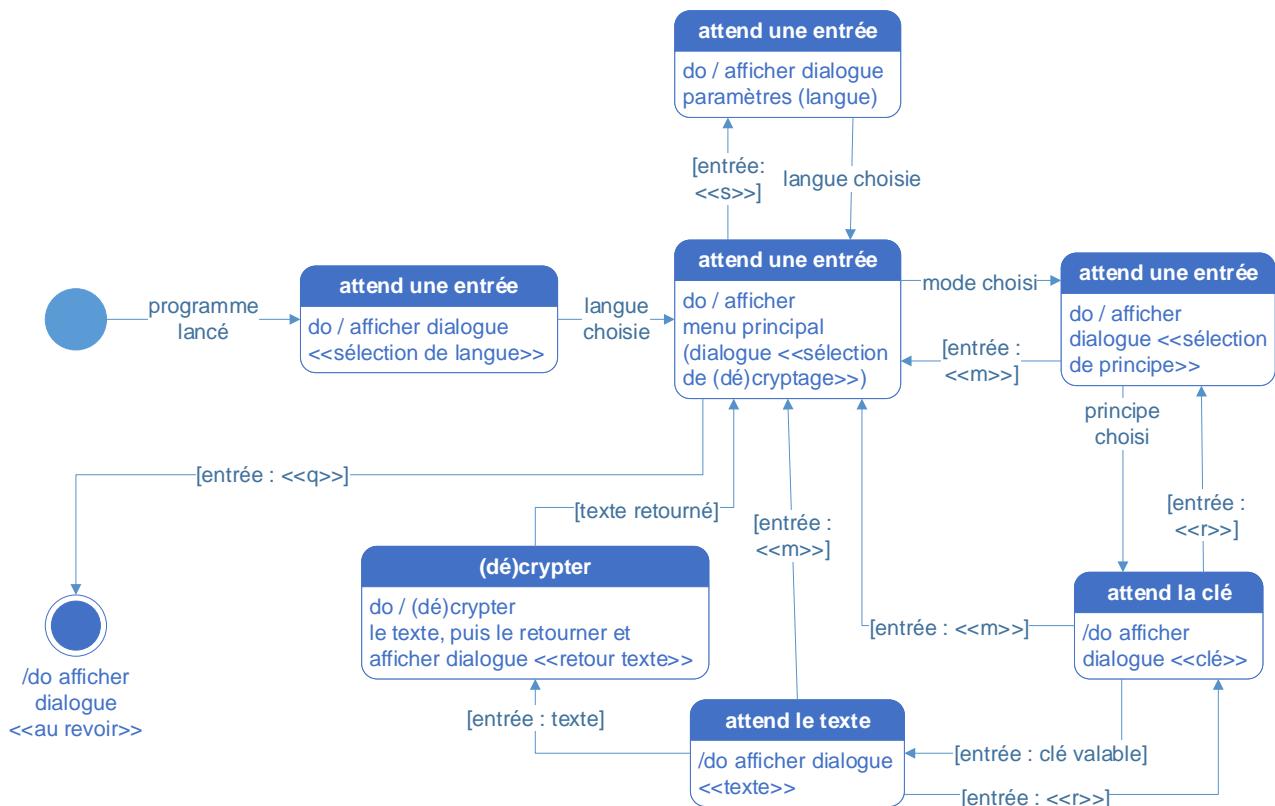


Fig. 16 – Diagramme états-transitions du programme

3.1.3 En détail et coupé en modules

- *Avant de commencer le (de)cryptage :*

On va créer une méthode qui formatera le texte (enlever les accents, les caractères spéciaux, les signes de ponctuation et les espaces, puis mettre tout en majuscule).

- *(Dé)Cryptage César :*

On compte transformer les lettres du texte en nombres par le système unicode, ensuite on additionne la clé (nombre) aux nombres obtenus par le système. On reconvertis alors les lettres en caractères et retourne le texte. Pour le décryptage, il suffit de soustraire au lieu d'additionner.

- *(Dé)Cryptage Vigenère :*

Dans un premier temps on crée une matrice de 26 x 26 lettres contenant l'alphabet représentant le tableau de Vigenère et une matrice à deux lignes pour le texte et la clé. Dans un deuxième temps on prend le texte et on insère tour à tour les caractères individuels dans la première ligne de la matrice et la clé dans la deuxième. Chaque lettre du texte doit être attribuée à un caractère de la clé (mot) ce qui est permis en répétant le mot tant qu'ils restent des lettres du texte. Le cryptage est fait caractère par un caractère. Pour trouver la lettre cryptée on regarde en premier la lettre du texte et on cherche la colonne de la matrice qui appartient à la lettre et on la mémorise. Par la suite on considère le caractère de la clé auquel la lettre du texte est attribuée et on cherche la ligne de la matrice qui appartient au caractère. Dès qu'on a trouvé les deux, la lettre de la matrice qui est enregistré sur cette case est la lettre cryptée laquelle est mémorisée dans une chaîne de caractère. Lorsqu'on a crypté toutes les lettres du texte, on retourne la chaîne de caractère qui est le texte crypté. Pour le décryptage, on possède la matrice à deux lignes avec le texte crypté sur une ligne et la clé répétée sur l'autre. Pour chaque lettre du texte crypté, on parcourt dans le tableau de Vigenère (matrice) la ligne correspondant à la lettre de la clé répétée et lorsqu'on trouve la lettre cryptée cherchée on remonte la ligne pour obtenir le caractère décrypté correspondant à cette lettre. De la même manière que pour le cryptage on mémorise les lettres obtenues dans une chaîne de caractère ce qui permet d'avoir au final le message décrypté.

- *(Dé)Cryptage Enigma :*

Pour commencer nous créons cinq listes : La première correspond au tableau de permutation, les trois suivantes aux rotors au milieu et la dernière au réflecteur. Au début de la programmation on choisit un décalage fixe ; si possible on essayera plus tard de changer le décalage automatiquement. Pour les connecteurs, il y en aura dix ainsi 20 lettres seront connectées entre elles et six qui ne le seront pas. Nous allons créer une fonction qui, si la lettre est connectée, va retourner la lettre associée. On appelle ensuite des fonctions qui exécutent les décalages au niveau des trois rotors et après on utilise la fonction correspondant au réflecteur pour permuter les lettres. On repasse finalement à nouveau par les rotors et on obtient une certaine lettre. Si la lettre est connectée à une autre, on retourne l'autre lettre, sinon on garde la même, celle-ci est alors la lettre (dé)cryptée.

3.1.4 Structure du programme

main_module	screen_module	tests
<pre>normalise_letter(x) format_text(text) run()</pre>	<pre>show_start_ask_language() show_language_settings() show_main_menu(english: Boolean) show_principles(english: Boolean, encrypting: Boolean) show_ask_key(english: Boolean, principle: char) show_ask_text(english: Boolean) show_treated_text(english: Boolean, encrypting: Boolean) show_help_principles(english: Boolean) show_quit_message(english: Boolean)</pre>	<pre>test_caesar_encryption() test_caesar_decryption() test_create_table_of_vigenere() test_vigenere_encryption() test_vigenere_decryption() test_enigma() test_format_and_normalise() test_run() run_all_tests()</pre>
<pre>screen_constants</pre>		
<pre>START_ASK_LANGUAGE LANGUAGE_SETTINGS ENGLISH_MAIN_MENU ENGLISH_PRINCIPLES_ENCRYPTING ENGLISH_PRINCIPLES_DECRYPTING ENGLISH_ASK_KEY_CAESAR ENGLISH_ASK_KEY_VIGENERE ENGLISH_ASK_KEY_ENIGMA ENGLISH_ASK_TEXT ENGLISH_ENCRYPTED_TEXT ENGLISH_DECRYPTED_TEXT ENGLISH_HELP_PRINCIPLES ENGLISH_QUIT_MESSAGE FRENCH_MAIN_MENU FRENCH_PRINCIPLES_ENCRYPTING FRENCH_PRINCIPLES_DECRYPTING FRENCH_ASK_KEY_CAESAR FRENCH_ASK_KEY_VIGENERE FRENCH_ASK_KEY_ENIGMA FRENCH_ASK_TEXT FRENCH_ENCRYPTED_TEXT FRENCH_DECRYPTED_TEXT FRENCH_HELP_PRINCIPLES FRENCH_QUIT_MESSAGE</pre>	<pre>text_module</pre>	
	<pre>encrypt_caesar(text: string, key: int) decrypt_caesar(text: string, key: int) create_vigenere_table() create_table_text_key(text: string, key: string) encrypt_vigenere(text: string, key: string) decrypt_vigenere(text: string, key: string) create_initial_list() search_index(initial_list: list, letter: char) plugboard(letter: char) permutation_reflector(letter: char) shift_first_rotor(return_path: Boolean, index: int, offset: int) shift_second_rotor(return_path: Boolean, index: int, offset: int) shift_third_rotor(return_path: Boolean, index: int, offset: int) enigma(text: string, key: string)</pre>	

Fig. 17 – Structure du programme en modules

Explications :

- **main_module :**
 - **normalise_letter(x) :**
normalise une lettre si elle est par exemple un caractère spécial
 - **format_text(text) :**
formate le texte, c'est-à-dire le texte donné est mis en majuscules, les virgules, espaces, chiffres, symboles, etc. sont enlevés et les caractères spéciaux sont normalisés
 - **run() :**
démarrer le programme et contrôle son déroulement
- **screen_constants :**
 - **START_ASK_LANGUAGE :**
dialogue de démarrage (voir fig. 1)
 - **LANGUAGE_SETTINGS :**
dialogue des paramètres de langue (voir fig. 3 et 4)

- **ENGLISH_MAIN_MENU** et **FRENCH_MAIN_MENU** :
menu principal (voir fig. 2 et ??)
- **ENGLISH_PRINCIPLES_ENCRYPTING** et
FRENCH_PRINCIPLES_ENCRYPTING :
dialogue de sélection du principe de cryptage (voir fig. ??, ??, 5 et 6)
- **ENGLISH_PRINCIPLES_DECRYPTING** et
FRENCH_PRINCIPLES_DECRYPTING :
dialogue de sélection du principe de décryptage (voir fig. ??, ??, 5 et 6)
- **ENGLISH_ASK_KEY_CAESAR** et **FRENCH_ASK_KEY_CAESAR** :
dialogue de demande de la clé pour (dé)crypter par César (voir fig. ??, ??, ??, 7, 8 et 9)
- **ENGLISH_ASK_KEY_VIGENERE** et **FRENCH_ASK_KEY_VIGENERE** :
dialogue de demande de la clé pour (dé)crypter par Vigenère (voir fig. ??, ??, ??, 7, 8 et 9)
- **ENGLISH_ASK_KEY_ENIGMA** et **FRENCH_ASK_KEY_ENIGMA** :
dialogue de demande de la clé pour (dé)crypter par Enigma (voir fig. ??, ??, ??, 7, 8 et 9)
- **ENGLISH_ASK_TEXT** et **FRENCH_ASK_TEXT** :
dialogue de demande du text à (dé)crypter (voir fig. ?? et 10)
- **ENGLISH_ENCRYPTED_TEXT** et **FRENCH_ENCRYPTED_TEXT** :
dialogue du texte crypté (voir fig. ??, ??, 11 et 12)
- **ENGLISH_DECRYPTED_TEXT** et **FRENCH_DECRYPTED_TEXT** :
dialogue du texte décrypté (voir fig. ??, ??, 11 et 12)
- **ENGLISH_HELP_PRINCIPLES** et **FRENCH_HELP_PRINCIPLES** :
dialogue d'aide (voir fig. 14)
- **ENGLISH_QUIT_MESSAGE** et **FRENCH_QUIT_MESSAGE** :
message de sortie (voir fig. ?? et 13)
- *screen_module* :
 - **show_start_ask_language()** :
affiche le dialogue de démarrage (voir fig. 1)
 - **show_language_settings()** :
affiche les paramètres (voir fig. 3 et 4)
 - **show_main_menu(english)** :
affiche le menu principal (voir fig. 2 et ??)

- **show_principles(english, encrypting)** :
affiche le dialogue de sélection du principe (voir fig. ??, ??, 5 et 6)
- **show_ask_key(english, principle)** :
affiche le dialogue de demande de la clé pour (dé)crypter ce qui dépend de la choix du principe (César, Vigenère ou Enigma ; voir fig. ??, ??, ??, 7, 8 et 9)
- **show_ask_text(english)** :
affiche le dialogue de demande du text à (dé)crypter
(voir fig. ?? et 10)
- **show_treated_text(english, encrypting)** :
affiche le texte (dé)crypté (voir fig. ??, ??, 11 et 12)
- **show_help_principles(english)** :
affiche le dialogue de démarrage (voir fig. 14)
- **show_quit_message(english)** :
affiche le message de sortie (voir fig. ?? et 13)
- ***text_module*** :
 - **encrypt_caesar(text, key)** :
crypte le texte formaté par la méthode César
 - **decrypt_caesar(text, key)** :
décrypte le texte formaté par la méthode César
 - **create_vigenere_table()** :
TODO
 - **create_table_text_key(text, key)** :
TODO
 - **encrypt_vigenere(text, key)** :
crypte le texte formaté par la méthode Vigenère
 - **decrypt_vigenere(text, key)** :
décrypte le texte formaté par la méthode Vigenère
 - **create_initial_list()** :
TODO
 - **search_index(initial_list, letter)** :
cherche l'indice d'une lettre donnée dans une liste donnée
 - **plugboard(letter)** :

modélise le tableau de permutation, 10 lettres sont associées à dix autres lettres. Si la lettre donnée est associée à une autre lettre celle-ci va être retournée. Il y a six lettres qui ne sont associées à aucune autre lettre.

- **permutation_reflector(letter : char)** :
modélise le réflecteur ; même principe que plugboard(letter), mais cette fois-ci toutes les lettres sont associées à une autre lettre
 - **shift_first_rotor(letter : char)** :
modélise le premier rotor, exécute le décalage des lettres par indices et retourne la lettre correspondante
 - **shift_second_rotor(letter : char)** :
modélise le deuxième rotor, exécute le décalage des lettres par indices et retourne la lettre correspondante
 - **shift_third_rotor(letter : char)** :
modélise le troisième rotor, exécute le décalage des lettres par indices et retourne la lettre correspondante
 - **enigma(text, key)** :
(dé)crypte le texte formaté par la méthode Enigma
- **tests :**
- **test_caesar_encryption()** :
TODO
 - **test_caesar_decryption()** :
TODO
 - **test_create_table_of_vigenere()** :
TODO
 - **test_vigenere_encryption()** :
TODO
 - **test_vigenere_decryption()** :
TODO
 - **test_enigma()** :
TODO
 - **test_format_and_normalise()** :
TODO
 - **test_run()** :
TODO

- **run_all_tests()** :
TODO

4 Annexe : Code du programme