

But

La machine *enigma* fût intensément utilisée pour transcoder des messages secrets en particulier pendant la deuxième guerre mondiale par les allemands. Afin de déchiffrer un message, il est nécessaire d'avoir exactement les mêmes configurations entre les différentes machines. Ces configurations changeaient tous les jours.

Alan Turing développa une machine « bombe » permettant de cracker ces paramètres et ainsi décoder les messages ennemis. Ceci reste un véritable exploit compte tenu de la technologie du moment.

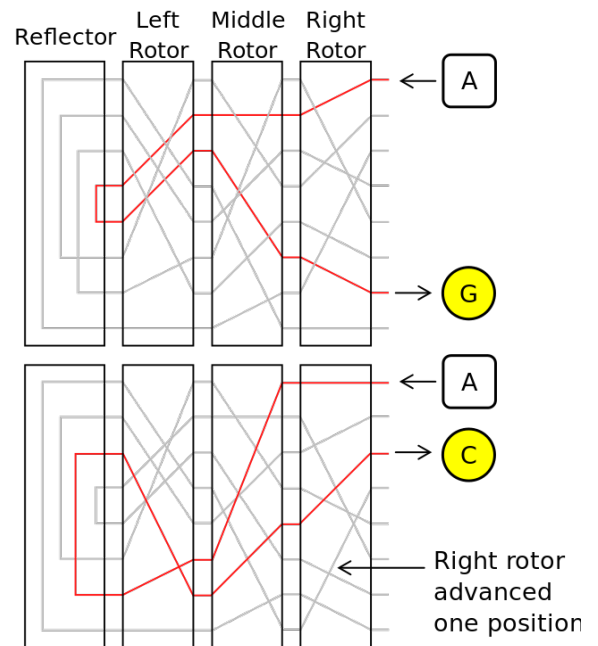
Pourtant le fonctionnement de la machine *enigma* est relativement simple mais offrait un nombre considérable de possibilités.

Ce laboratoire vise à reproduire cette machine.

Avant de continuer, il est utile de consulter ces liens

- Vidéo https://www.youtube.com/watch?v=mcX7iO_XCFA
- Simulateur <https://cryptii.com/pipes/enigma-machine>
- Wiki https://en.wikipedia.org/wiki/Enigma_rotor_details
- Exemple <https://www.codesandciphers.org.uk/enigma/example1.htm>

Implémenter les classes nécessaires afin d'implémenter la machine *enigma* avec les codes disponibles suivants. Pour simplifier ce développement, nous ignorons le *plugboard*.



Component	Wiring	Id	Notch
ENTRY	ABCDEFGHIJKLMNOPQRSTUVWXYZ		
Rotor	EKMF LGDQVZNTOWYHXUSPAIBRCJ	I	R
	AJDKSIRUXBLHWTMCQGNPYFVOE	II	F
	BDFHJLCPRTXVZNYEIWGAKMUSQO	III	W
	ESOV PZJAYQUIRHXLNFTGKDCMWB	IV	K
	VZBRGITYUPSDNHLXAWMJQOFECK	V	A
Reflector	EJ MZALYXVBWFCRQUONTSP IKHGD	UKW-A	
	YRUHQSLDPXNGOKMIEBFZCWVJAT	UKW-B	
	FVPJIAOYEDRZXWGCTKUQSB NMHL	UKW-C	

Sur ces bases, écrire un programme pour décoder le message

MDXMDAORNSLZBJTCDSABGHLVWA

... avec les configurations

Component	Id	Position
Rotor - LEFT	II	C
Rotor - MIDDLE	IV	K
Rotor - RIGHT	I	M
Reflector	UKW-B	

A faire

Par les différents fichiers et classes, vous devez mettre à disposition de quoi :

- créer un objet de type *Enigma* en passant les rotors et le réflecteur utilisés
- changer le réflecteur
- changer un rotor
- changer la position d'un rotor
- convertir un caractère
- convertir une chaîne de caractères
- choisir d'afficher les informations de cheminement (debug) tant pour les constructeurs que pour les conversions (**voir exemple en dernière page**)

Contraintes

- Lire les documentations proposées et liens afin de bien comprendre le sujet
- Utiliser au mieux la théorie et les éléments vus à ce jour
- Ne rien utiliser qui n'est pas encore étudié en théorie (ie héritage ...)
- Répartir les différentes classes dans des fichiers distincts

Temps à disposition : 10 périodes

Configuration initiale

LEFT rotor

```
rotor id : II
entry   : ABCDEFGHIJKLMNOPQRSTUVWXYZ
wiring  : AJDKSIRUXBLHWTMCQGZNPYFVOE
position : C
notch   : F
```

MIDDLE rotor

```
rotor id : IV
entry   : ABCDEFGHIJKLMNOPQRSTUVWXYZ
wiring  : ESOPVZJAYQUIRXLNFTGKDCMWB
position : K
notch   : K
```

RIGHT rotor

```
rotor id : I
entry   : ABCDEFGHIJKLMNOPQRSTUVWXYZ
wiring  : EKMFLGDQVZNTOWYHXUSPAIBRCJ
position : R
notch   : R
```

Reflector

```
reflector : UKW-B
entry     : ABCDEFGHIJKLMNOPQRSTUVWXYZ
wiring    : YRUHQS LDPXNGOKMIEBFZC W VJAT
```

Exemple de codage de la lettre 'B'

in out

```
rotor id : I
entry   : 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
wiring  : E K M F L G D Q V Z N T O W Y H X U S P A I B R C J
notch   : R
position : S
result  : P <= B
```

```
rotor id : IV
entry   : 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
wiring  : E S O V P Z J A Y Q U I R X L N F T G K D C M W B
notch   : K
position : L
result  : E <= P
```

```
rotor id : II
entry   : 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
wiring  : A J D K S I R U X B L H W T M C Q G Z N P Y F V O E
notch   : F
position : D
result  : U <= E
```

```
reflector : UKW-B
entry     : 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
wiring    : Y R U H Q S L D P X N G O K M I E B F Z C W V J A T
result    : U => C
```

```
rotor id : II
entry   : 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
wiring  : A J D K S I R U X B L H W T M C Q G Z N P Y F V O E
notch   : F
position : D
result  : C => M
```

```
rotor id : IV
entry   : 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
wiring  : E S O V P Z J A Y Q U I R X L N F T G K D C M W B
notch   : K
position : L
result  : M => M
```

```
rotor id : I
entry   : 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
wiring  : E K M F L G D Q V Z N T O W Y H X U S P A I B R C J
notch   : R
position : S
result  : M => K
```