

## Ave mes niewbus Corticus

0 lib externe :) 100%algo

Objectif du projet créer une application qui va encoder et décoder des messages. La même application sera utilisée pour les autres exos

Le projet doit être bien structurer en gros des classes... sinon la bulle et le fouet

Vous devez proposer une IHM qui permettra de switcher d exo.

0 code logique liée au chiffrement dans la classe IMH (for / foreach ...)

### Exo 1 L'algorithmme de Caesar.

Le principe est de décaler de 3 lettres vers la droite un caractère : ( a => d , B => E, z => c ...)

Demander à l'utilisateur un choix d'action => Decode / Encode"

Lire et proposer la réponse a l'utilisateur

Les char non-alphabétique ne sont pas transposé

La classe Cesar doit implémenter IEncryptionMethod

IEncryptionMethod definition :

void encryptChar(char plainchar)

void decryptChar(char cipherchar)

void reset();

Version moderne :

A chaque fois que la fonction encryptChar est exécuté le décalage augmente de 1 pour le prochain char

La fonction reset remet l'index a 3

## Exo2 Chiffrement de message One Time pad

Chiffrement base sur une clé aléatoire.

Le chiffrement consiste à combiner le message en clair avec une clé présentant les caractéristiques très particulières suivantes :

- Les caractères composant la clé doivent être choisis de façon totalement aléatoire.
- Chaque clé, ou « masque », ne doit être utilisée qu'une seule fois.

ex :

- A bah zour !
- avec la cle "bD"
- B cdi arvu !

La classe OneTime doit implémenter IEncryptionMethod

OneTime doit avoir comme constructeur :

OneTime(string key , string key)

la fonction reset remet key a sa position initial

## Exo3

Utiliser les classes afin de créer une classe static Encryption

```
static void EncodeString(IEncryptionMethod method, string toEncode)
```

```
static void DecodeString(IEncryptionMethod method , string toDecode)
```

usage :

```
var c = new Ceasar();
```

```
Encryption.EncodeString(c, "coucou petite perruche");
```

```
Encryption.DecodeString(c, 'blabla');
```