Université Paris Diderot
L3 informatique
Sécurité 2017-2018

Chiffrement par substitution

Les textes considérés sont composés des lettres romaines minuscules (`'abcdefghijklmnopqrstuvwxyz'`) ainsi que d'autres caractères (`' '`, `'-'`,`','`, `'1'`,...). Cependant pour simplifier le décryptage on supposera que ces autres caractères ne sont pas chiffré. (Dans un texte, les lettres majuscules sont remplacées par les les lettres minuscules correspondantes.)

Dans le chiffre de César, le texte chiffré s'obtient en remplaçant chaque lettre du texte en clair par une lettre à distance fixe dans l'ordre de l'alphabet. Par exemple si cette distance est 3, `'abd'` sera chiffré en `'deg'`.

Dans le chiffre par permutation, étant donnée une permutation des lettres σ, chaque lettre x est chiffrée par $\sigma(x)$.

Le chiffre de Vigenère réalise un décalage différent suivant la position de la lettre dans le texte. Le décalage est définie à l'aide d'un mot-clef. Dans ce mot-clef, chaque lettre correspond à un décalage. Ainsi si le mot-clef est `bal` (correspondant au décalage de 1, de 0 et de 11) pour le texte à chiffrer "`bonjour`", le `'b'` sera décalé de 1, le `'o'` ne sera pas décalé, `'n'` sera décalé de 11, `'j'` sera décalé de 1, et ainsi de suite en réutilisant cycliquement le mot-clef. On obtient ainsi "`coykofs`".

1 Chiffrage et déchiffrage

Dans cette partie on devra écrire des classes java permettant de faire le chiffrage et le déchiffrage pour ces chiffres alphabétiques. Pour cela:

- On commencera par définir une interface avec des méthodes chiffrer et dechiffrer pour le chiffrage et déchiffrage (pour un caractère, pour une String et pour tout autre type de données souhaitable).
- Ensuite on créera des classes ad hoc qui implémentent cette interface pour les trois types de chiffres décrits ci-dessus.

Vous écrirez deux applications chiffre et dechiffre, une pour le chiffrage la deuxième pour le déchiffrage, ayant comme premier paramètre le type de chiffrage (c/p/v pour César, permutation et Vigenère), comme deuxième paramètre la "clé" du chiffrage (valeur du décalage pour César, la permutation pour un chiffre par permutation, le mot-clef pour Vigenère), comme troisième un fichier contenant le texte à chiffrer ou à déchiffrer et qui affichent sur la sortie standard le texte chiffré ou déchiffré suivant les cas.

2 Décryptage

Le but de cette partie est d'essayer de déchiffrer ces différents textes chiffrés sans la clef. Pour cela on suppose que l'on dispose d'un texte chiffré qui était initialement en anglais (suffisamment long) dont on connait la méthode de chiffrement mais dont on ne connait pas la clé (décalage pour Cesar, la permutation pour le chiffre par permutation ou le mot-clef si c'est le chiffre de Vigenere (mais dans ce cas on suppose la longueur du mot-clef connue)).

On peut disposer d'une liste des mots anglais (accessible sous moodle) ainsi que des fréquences des lettres en anglais:

Lettre	Fréquence	Lettre	Fréquence
a	8.08 %	n	7.38 %
b	1.67 %	o	7.47 %
c	3.18 %	p	1.91 %
d	3.99 %	q	0.09 %
e	12.56 %	r	6.42 %
f	2.17 %	s	6.59 %
g	1.80 %	t	9.15 %
h	5.27 %	u	2.79 %
h	7.24 %	v	1.00 %
j	0.14 %	w	1.89 %
k	0.63 %	x	0.21 %
l	4.04 %	y	1.65 %
m	2.60 %	z	0.07 %

1. Du fait du nombre limité de ses clefs le chiffre de César est facile à décrypter. On définira et implémentera trois méthodes:
   (a) La première basée sur la connaissance d'un mot: on sait que le mot est dans le texte en clair (mais on ne sait pas à quelle position). En cas d'ambiguité on pourra utiliser la liste des mots du français pour lever cette ambiguité.
   (b) La deuxième est basée sur les fréquences: à partir de l'analyse des fréquences des lettres on déterminera le décalage.
   (c) Par force brute on essaiera tous les décalages jusqu'à obtenir un texte dont tous les mots (ou presque) sont dans le dictionnaire.
2. Pour le chiffrage de Vigenère, on suppose que l'on connaît la taille du mot-clef, définir des décryptages qui pourront utiliser les classes définies dans la question précédente.

   Pour ceux qui veulent plus: Que peut-on faire si on ne connaît pas la taille de la clef?
3. Pour le codage par permutation, définir un décryptage partiel basé sur les fréquences des lettres.

   Pour ceux qui veulent plus, réaliser un décryptage complet.

Vous définirez une application `decrypt` ayant comme premier paramètre le type de codage, comme deuxième paramètre un fichier contenant le texte à décrypter et qui affiche sur la sortie standard le texte décrypté. Dans le cas d'un décryptage de Cesar, un troisième paramètre donnera la stratégie (`1/2/3`) et pour la stratégie 1 un quatrième paramètre donnera le mot qui appartient au texte en clair. Dans le cas d'un décryptage de Vigenère un troisième paramètre donnera la taille du mot-clef.

Vous afficherez sur la sortie erreur une évaluation du temps d'exécution.

```
long startTime = System.currentTimeMillis();
      //          votre méthode
long endTime = System.currentTimeMillis();
System.err.println("Temps de xxxxx:" + (endTime-startTime) + "ms");
```

3 Travail à rendre

Vous pouvez réaliser ce travail par groupe d'au plus 3 personnes. Vous devez rendre sous moodle **un travail par groupe** pour le 6 avril 2018.

1. vos programmes java

2. un court rapport décrivant vos choix d'algorithme et de programmation et indiquant **précisément** comment exécuter vos applications.

3. le certificat contenant votre clef publique (format RSA),

4. un jar signé avec la clef correspondante contenant les .class de l'application (on doit pouvoir executer votre application sous la forme `java -jar ...`).

Pour la soutenance vous préparerez une démonstration avec des fichiers de trois tailles différentes (100, 500 et 2 000 caractères (environ)).