Devoir à la Maison n°1

Cryptage RSA

Le chiffrement RSA est un procédé de cryptage de données couramment utilisé pour les transmissions par internet. Son nom est composé des initiales des trois informaticiens qui l'ont défini en 1977 : Ronald Rivest, Adi Shamir et Leonard Adleman.

Le principe est le suivant :

- Les données sont numérisées, c'est-à-dire converties en entiers naturels.

  Tous ces entiers naturels sont supposés inférieurs à un entier n fixé, appelé *module de chiffrement*.
- Un entier est crypté en appliquant la fonction $F_{n,c}$ qui à un entier naturel a associe le reste de la division euclidienne de a^c par n, où c est un entier naturel, appelé *clef publique*.
- Pour décrypter il suffit d'appliquer la fonction $F_{n,d}$ c'est-à-dire la même fonction mais en remplaçant c par d, entier naturel appelé *clef privée*.
- Ensuite on convertit les entiers au format de départ, par exemple du texte.

L'utilisateur qui souhaite recevoir des donnés cryptées communique le module de chiffrement et la clef publique (n, c) à tous les utilisateurs souhaitant lui envoyer des données.

Il ne communique pas la clef privée d, c'est elle qui lui permet de décrypter les données.

Le but de ce devoir est de programmer et de tester ce cryptage.

Le langage utilisé doit être Python, il faut donc tester les fonctions sur un ordinateur sur lequel une distribution de Python est installée. Consulter pour ceci le site `prepabellevue.org`, rubrique `PCSI2/Informatique/Installation de Python`.

Les questions précédées du symbole ▷ doivent être faites sur votre ordinateur, mais ne nécessitent pas de rédaction sur votre copie. Les questions encadrées doivent être rédigées.

Les programmes et fonctions demandés doivent être manuscrits, les impressions de listings ne sont pas acceptées.

Il est inutile au sein d'une fonction de vérifier que les variables d'entrée satisfont les conditions nécessaires pour que le calcul soit défini.

Par contre il est important de faire figurer la documentation des fonctions, et de commenter les lignes de code.

Les tests sont parfois explicitement demandés, sinon il n'est pas nécessaire de les faire figurer sur la copie.

Partie A. Codage des caractères

Chaque caractère est codé par un entier, c'est le codage ASCII étendu. Par exemple les lettres minuscules `a` à `z` sont codées de 97 à 122.

En python la fonction `ord` donne le code ASCII d'un caractère, tandis que la fonction `chr` donne le caractère codé par l'argument.

```
>>> ord("a")
97
```

```
>>> chr(122)
'z'
```

▷ Obtenir les caractères codés par les entiers de 32 à 127 :

```
for k in range(32,128):
    print(k,chr(k))
```

1. Écrire une fonction `Numerisation` qui reçoit une chaine de caractères et qui renvoie la liste des codes ASCII de ses lettres.

Par exemple `Numerisation("Test")` doit renvoyer `[84,101,115,116]`.

▷ Tester l'exemple ci-dessus.

2. Écrire une fonction `ConversionStr` qui reçoit une liste d'entiers et qui renvoie la chaine de caractères qu'elle code.

Par exemple `ConversionStr([84,101,115,116])` doit renvoyer `"Test"`.

3. Quelle chaine de caractères est codée par `[66,114,97,118,111,32,33]` ?

Partie B. Programmation du cryptage

4. Écrire une fonction `F(a,n,c)` qui à l'entier a associe le reste de la division euclidienne de a^c par n.

▷ Tester l'instruction `F(37,199,25)`, elle doit renvoyer 141.

5. Écrire une fonction `RSA(L,n,c)` qui reçoit une liste d'entiers et qui renvoie la liste des éléments $F(a,n,c)$ où a parcourt la liste `L`.

▷ Tester l'instruction `RSA([0,1,2,3,4],199,85)`, elle doit renvoyer `[0,1,196,149,9]`.

6. Écrire une fonction `Cryptage(S,n,c)` qui reçoit une chaine de caractères `S` et qui renvoie le cryptage RSA de `S` avec n pour module de chiffrement et c pour clef.

▷ Tester l'instruction `Cryptage("Test",199,85)`, elle doit renvoyer `"KS|o"`.

7. Que renvoie l'instruction `Cryptage("KS|o",199,7)` ? Que peut-on en conclure ?