UNIVERSITÉ DE RENNES 1

M1 2016 - 2017

SECU **TP3**

Exercice 1 -

Le but de l'exercice est d'implémenter un générateur de clés RSA ainsi que les fonctions de chiffrement et de déchiffrement. Le TP est à faire en python.

1. Ecrivez une fonction qui prend en entrée trois entiers a et n et m et qui renvoie $a^n \mod m$.
2. Ecrivez une fonction qui prend en entrée deux entiers a et p et qui renvoie **true** si $a^{p-1} = 1 \mod p$ et **false** sinon. Il s'agit du test de primalité de Fermat.
3. Ecrivez une fonction qui prend en entrée un nombre entier p et qui renvoie **true** si et seulement si $2^{p-1} = 3^{p-1} = 5^{p-1} = 7^{p-1} = 1 \mod p$. Il s'agit du test de primalité effectué dans PGP. Dans la suite on utilisera exclusivement cette fonction pour tester la primalité d'un nombre entier.
4. Ecrivez une fonction qui prend en entrée un nombre entier n et qui renvoie un nombre premier de n bits aléatoire.
5. Ecrivez un générateur de clés RSA. La sortie est un tuple $\{e, d, N, p, q\}$.
6. Générez une paire de clés RSA et affichez la.
7. Affichez aussi $\varphi(N)$ et $N - \varphi(N)$. Que constatez-vous ?
8. Ecrivez les fonctions de chiffrement et de déchiffrement RSA.
9. Testez ces fonctions.

Exercice 2 -

1. Implémentez le crible d'Ératosthène.
2. Quelle est la complexité de cet algorithme ?
3. Affichez les 100 premiers nombres premiers.
4. Comparez les résultats avec les tests de primalité de l'exercice précédent.