CRYPTOGRAPHIE APPLIQUÉE

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

D'après les cours dispensés à l'ENS par Anca Nitulescu

A. CHIFFREMENTS PRÉ-INFORMATIQUE

Sauf mention contraire, on ne considérera dans la suite que les caractères majuscules de A à Z. Par convention, sauf mention contraire, le rang de la lettre A sera 0, et le rang de Z sera 25.

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Chiffre de César

C'est probablement le chiffre de **substitution** le plus ancien, attribué à Jules César ; il consiste à décaler dans l'alphabet chaque lettre du message en clair d'une valeur fixe ***k***.

Question 1

Écrivez un programme qui permet de chiffrer le texte en clair ***m*** à l'aide de la clé ***k***.

Question 2

Écrivez un programme qui demande un texte chiffré ***s*** et qui affiche les 25 possibilités de texte clair.

Question 3

Quel est le texte clair correspondant au message MILOBCOMZYVIDOMRKXQOBC ?

Chiffre affine

Dans ce code, on utilise comme clé deux entiers ***a*** et ***b***. On prend le rang r_i de chaque lettre du message en clair auquel on applique la *formule* $y = (a \times r_i + b) \bmod 26$ qui donne le rang ***y*** de la lettre chiffrée.

Lorsque la valeur de ***a*** est égale à 1, on retrouve le chiffre de César.

Déchiffrement et notions d'arithmétique modulaire

Pour déchiffrer le message, il faut être capable de trouver l'antécédent de ***y*** calculé précédemment, ce qui va nécessiter quelques notions en arithmétique modulaire.

Soient $a, n, m \in Z\ (m > 0)$. On note $a \equiv n \bmod m$ si m divise $(a - n)$

Ainsi $26 \equiv 2 \bmod 24$ car $\frac{26-2}{24} \in Z$

On définit l'anneau $Zm=Z/mZ$ constitué de l'ensemble {0, 1, 2, …, m−1} et de deux opérations :

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

Addition : à deux restes a et b, on associe le reste de a+b modulo n.

$$a+b \equiv c \ mod \ m$$

Multiplication : à deux restes a et b, on associe le reste de a.b modulo n.

$$a.b \equiv d \ mod \ m$$

Table d'addition dans ℤ/6ℤ

+	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	2	3	4	5	0
2	2	3	4	5	0	1
3	3	4	5	0	1	2
4	4	5	0	1	2	3
5	5	0	1	2	3	4

Table de multiplication dans ℤ/6ℤ

×	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	4	0	2	4
3	0	3	0	3	0	3
4	0	4	2	0	4	2
5	0	5	4	3	2	1

Une notion très importante, qui sera utile, entre autre, pour le déchiffrement affine, est l'inverse modulaire :

Soient a et m, on cherche a^{-1} tel que $a.a^{-1} \equiv 1 \ mod \ m$

Ainsi par exemple pour $a = 2$ et $m = 9$, on a $2^{-1} \equiv 5 \ mod \ 9$ car $2.5 = 10 \equiv 1 \ mod \ 9$

Question 4

Trouvez l'inverse de a=4 avec m=9

Question 5

Même question avec a=6 et m=9

Existence et unicité de l'inverse

Le théorème de Bachet-Bézout prouve que l'inverse de $a \ mod \ m$ existe si et seulement si (a,)=1, c'est-à-dire que a et m doivent être premiers entre eux. Dans le cas où cet inverse existe, alors il est **unique** et il est calculable grâce à l'algorithme d'Euclide étendu ou au théorème d'Euler.

Question 6

Codez l'algorithme d'Euclide étendu.

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

Question 7

Écrivez un programme de déchiffrement affine avec ***a*** et ***b*** en paramètres.

B. CHIFFREMENT RSA

Il s'agit d'un chiffrement asymétrique avec une clé publique et une clé privée. Pour créer ses clés, Alice procède de la façon suivante :

- Alice choisit deux nombres premiers ***p*** et ***q*** et calcule le produit ***n*** = ***p*** . ***q***.
- Alice choisit également un entier ***e*** premier avec ϕ(n) = (***p*** − 1)(***q*** − 1).
- Alice publie sa clé publique : (***n***, ***e***)
- Alice calcule sa clé privée ***d*** telle que tel que $d.e \equiv 1 \ mod\ \phi(n)$

Question 8

On considère les valeurs ***p*** = 53, ***q*** = 11 et ***e*** = 3. Calculez la valeur ***d*** de la clé privée.

Chiffrement par Bob

Bob souhaite envoyer un message chiffré à Alice, il connait sa clé publique, c'est-à-dire les valeurs ***n*** et ***e***. Il va transformer son message en remplaçant chaque lettre par son rang :

P	O	L	Y	T	E	C	H	A	N	G	E	R	S
15	14	11	24	19	04	02	07	00	13	06	04	17	18

Ensuite, en partant de **la droite**, Bob va décomposer le message en bloc de même taille. Dans cet exemple, on va prendre une taille de 3, on y reviendra plus tard.

Question 9

Complétez le tableau des blocs suivant :

Bob va ensuite calculer, pour chaque bloc B, le bloc C selon la formule : $C = B^e \pmod n$: cela constitue le message chiffré que Bob va envoyer à Alice.

Question 10

Complétez le message chiffré envoyé par Bob.

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

1	303								

Déchiffrement par Alice

Alice reçoit les blocs C chiffrés par Bob. Pour les déchiffrer, il lui suffit d'appliquer la formule suivante : $B = C^d \pmod n$

Question 11

Complétez le message déchiffré par Alice (faites le **calcul jusqu'au bout**).

1	514								

Question 12

Lors du chiffrement, on a choisi une taille de 3. Ce « choix » est-il judicieux ? Que proposez-vous ?

C. SIGNATURE À L'AVEUGLE

On est dans la même situation que précédemment, Alice dispose de clés RSA valides. Bob souhaite qu'Alice appose sa signature sur un message ***m***, sans qu'Alice ne prenne connaissance du message.

On suppose que le message ***m*** est un entier inférieur à ***n***. Bob choisit un entier ***k*** premier avec ***n*** et transmet à Alice l'entier $m' = mk^e \pmod n$.

Alice va alors apposer sa signature en effectuant le calcul $m'' = (m')^d \pmod n$ et va transmettre m'' à Bob.

- Bob peut alors calculer $s = m''.k^{-1} \pmod n$. Avec le couple (m,s), Bob dispose d'un message signé par Alice, c'est-à-dire qui vérifie $s^e \equiv m \bmod n$

Question 13

Codez l'algorithme précédent. On suppose qu'Alice a choisi p=5, q=11, et e=27. Vérifiez le fonctionnement de la signature sur deux exemples.

D. FONCTION DE HACHAGE

On va définir une fonction de hachage très simple (TTH) que l'on va pouvoir facilement coder ou calculer à la main. On part d'une valeur initiale (IV) égale à (0, 0, 0, 0) et on va la faire évoluer après différentes étapes. Ensuite on va découper le message à hacher en bloc de 16 caractères (on complètera avec des 'A' le dernier bloc, le cas échéant) et on va effectuer une série d'opérations :

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

Première étape

Les lettres de chaque bloc vont être disposées ligne par ligne dans un carré 4x4 et on va en prendre le rang. Par exemple, si le premier bloc est composé des lettres ABCDEFGHIJKLMNOP, on aurait les tableaux suivants :

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

On va effectuer la somme colonne par colonne, modulo 26 :

24	2	6	10

On va ensuite ajouter IV. Pour le premier bloc, cette somme reste inchangée.

Deuxième étape

Dans ce second temps, on effectue une permutation des valeurs : on décale la première ligne de une case vers la gauche, la deuxième de deux cases, la troisième de trois cases, et on inverse l'ordre de la dernière ligne :

1	2	3	0
6	7	4	5
11	8	9	10
15	14	13	12

On effectue à nouveau la somme colonne par colonne, modulo 26 :

7	5	3	1

On additionne cette somme à la précédente (modulo 26), et ce résultat devient la nouvelle IV du bloc suivant :

5	7	9	11

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

Une fois tous les blocs traités, on transforme le résultat obtenu en lettres, soit dans notre exemple **FHJL**.

Question 14

Codez cet algorithme, quelle est la signature du texte « BIENVENUEAPOLYTECHANGERS » ?

E. EXEMPLE DE PROTOCOLE DE VOTE ÉLECTRONIQUE

Nous allons voir un processus possible de vote électronique, avec différents acteurs. On suppose que le vote est représenté par une lettre.

Préparation du vote

Tout d'abord le **commissaire** au vote va créer une liste qui attribue à chaque électeur un code N1, et va faire parvenir ce code à chacun. On va supposer que ce code est composé de12 lettres, par exemple : AZFJ VDES VAOX.

Lorsqu'un électeur reçoit son code N1, il crée alors un code N2 de son choix dont il calcule l'empreinte (dans notre exemple la fonction de hachage TTH vue précédemment). Il renvoie alors au **commissaire** son N1 accompagné de l'empreinte de son N2.

Jour du scrutin

L'électeur contacte l'**administrateur** avec son code N1. Ce dernier vérifie que le code est valide auprès du **commissaire**. Une fois validé, l'électeur crée son bulletin de vote avec son choix suivi de son code N2. Par exemple, si le code N2 est ABCDEFGHIJKL et le vote Z, le bulletin de vote serait :

Z	A	B	C	D	E	F	G	H	I	J	K	L

L'électeur doit maintenant faire signer son vote par l'**administrateur**, mais sans que ce dernier ait connaissance du vote : on applique donc le protocole de signature à l'aveugle et l'électeur récupère alors son bulletin de vote signé.

L'électeur utilise la clé publique d'un nouvel intervenant, le **décompteur**, afin de chiffrer son bulletin de vote (c'est l'équivalent de mettre sous enveloppe) et obtient donc son vote chiffré **VC**.

Pour terminer son vote, l'électeur envoie alors à l'**anonymiseur** le couple (N1, VC) : l'**anonymiseur** vérifie auprès du **commissaire** la validité de N1. Si tout va bien, le **commissaire** note que N1 a voté, et l'**anonymiseur** enregistre le vote.

Dépouillement

Une fois les votes terminés, le **décompteur** déchiffre à l'aide de sa clé privée les bulletins stockés par l'**anonymiseur** et procède aux opérations suivantes :

- il vérifie l'authenticité du bulletin grâce à la clé publique de l'**administrateur**
- il transmet au **commissaire** le code N2 qui, en calculant l'empreinte, vérifie que ce code est valide.

Alain Godon <alain.godon@univ-angers.fr>

v 1.01

Question 15

Codez le protocole et testez-le.

Question 16

Quelles remarques / améliorations faites-vous ?