Faculté des sciences exactes et naturelles
Licence Informatique / INFO0603 : Compression et cryptographie
Année universitaire 2016-2017
Pascal Mignot

Université de Reims Champagne-Ardenne

Examen terminal du 18 mai 2017
Première session

Instructions :

- Le polycopié de cours est autorisé. Tout autre document (TDs, notes personnelles, livres, ...) est interdit.
- L'usage de la calculatrice est autorisé. Néanmoins, celle-ci ne devra pas avoir de fonctions communicantes (*i.e.* l'utilisation d'un téléphone portable comme calculatrice est interdit).
- Un résultat donné sans justification sera compté faux (*i.e.* joindre vos brouillons si nécessaire).

Notation : A_x signifie que le nombre A est écrit en base x. Par exemple, 101_2 est 101 en binaire (soit 5 en décimal), 12_{16} est le nombre hexadécimal 12 (soit 18 en décimal). Néanmoins, la base n'est pas précisée lorsqu'elle est évidente dans le contexte.

Exercice 1: Chiffre par décalage

Le message `HLZRZDVSZVETYRKZVSZVE` a été codé avec un chiffre mono-alphabétique par décalage. On rappelle les fréquences des lettres les plus fréquentes : e=17.5%, s=8,0%, a=7,6%, i=7,5%, t=7,2%, n=6,6%.

1. Donner deux méthodes permettant d'éviter de tester toutes les clefs.
2. Décoder le message en utilisant la première méthode.
3. Décoder le message en utilisant la seconde méthode.
4. Quel est le principal inconvénient de ce cryptosystème ?

Exercice 2: Chiffre de Vigenère

On souhaite effectuer la cryptanalyse du message suivant "`TERTRE TRES TERRESTRE`" codé avec un chiffre de Vigenère qui utilise un alphabet de 4 symboles $\{E, R, S, T\}$.

Ce message est constitué d'une suite de mots français utilisant les mêmes lettres (*i.e.* pour le code et le texte clair : $E = 0$, $R = 1$, $S = 2$, $T = 3$).

1. Calculer le chiffre de Vigenère pour la clef STR.
2. On voudrait faire de la cryptanalyse. Expliquer pourquoi le calcul des fréquences de lettres n'est d'aucune aide dans ce cas.
3. Appliquer le test de Kasiski sur des motifs de taille au moins 3 afin de déterminer la taille de la clef.
4. En quoi la connaissance de la longueur de la clef va-t-elle nous aider dans la cryptanalyse ?
5. Calculer l'index de coïncidence pour le langage. On supposera que la fréquence des lettres dans le langage est identique à leur fréquence d'apparition dans le texte clair.
6. Calculer les fréquences des lettres pour le troisième symbole de la clef.

7 Calculer l'indice de coïncidence de Friedmann avec deux des possibilités de symboles pour le troisième symbole de clef, dont le bon (par exemple, on effectuera le calcul pour A et R).

8 Comment déduit-on alors le troisième symbole de la clef à partir des indices de coïncidences.

Exercice 3: cryptosystème produit d'un chiffre RSA

1 Que peut-on espèrer en effectuant le produit de deux cryptosystèmes RSA ?

2 On considère les deux cryptosystèmes RSA suivants : $S_1 = (n, p, q, a_1, b_1)$ et $S_2 = (n, p, q, a_2, b_2)$.

a. Le cryptosystème $S_1 \times S_2$ obtenu est-il commutatif ?
b. Le cryptosystème $S_1 \times S_2$ obtenu est-il idempotent ?
c. Si le cryptosystème obtenu à la question précédente est idempotent, donner son chiffre RSA équivalent.

3 On considère les deux cryptosystèmes RSA suivants : $T_1 = (n_1, p_1, q_1, a_1, b_1)$ et $T_2 = (n_2, p_2, q_2, a_2, b_2)$.

a. Le cryptosystème $T_1 \times T_2$ obtenu est-il commutatif ?
b. Le cryptosystème $T_1 \times T_2$ obtenu est-il idempotent ?
c. Si le cryptosystème obtenu à la question précédente est idempotent, donner son chiffre RSA équivalent.

4 Conclure sur l'utilisation d'un cryptosystème RSA produit.

Exercice 4: Schéma de Feistel

On définit trois fonctions prenant un paramètre t de 2 bits :

- $f_1(t)$ = met le premier bit de t à 0.
- $f_2(t)$ = inversion des bits de t.
- $f_3(t)$ = met le dernier bit de t à 1.

1 Expliquer pourquoi, même si les fonctions f_i ne sont pas bijectives (= peuvent détruire de l'information), un schéma de Feistel est inversible.

2 On considère un alphabet de 4 symboles $\{E, R, S, T\}$. Donner le codage à taille fixe associé.

3 Chiffrer le bloc "STRESSEE" en utilisant le schéma de Feistel à 3 ronde $\Psi^3(f_1, f_2, f_3)$ sur des blocs de 4 bits (i.e. 2 blocs de 2 bits).

4 Décoder le message "SSTRTSSS" codé avec le schéma de Feistel à 3 ronde $\Psi^3(f_1, f_2, f_3)$ sur des blocs de 4 bits (i.e. 2 blocs de 2 bits).

5 Donner alors le résultat du chiffrement des blocs du texte clair obtenu à la question 3 par un mode opérateur ECB.

6 Quel serait l'intérêt d'utiliser plutôt le mode CBC ? On prendra IV $= 01_2$.

Exercice 5: RSA

On souhaite construire un cryptosystème RSA (n, p, q, a, b).

1 Quelles sont les propriétés que doivent respecter n dans ce cas ?

2 Rappeler la définition d'un nombre premier, et montrer que si l'on veut tester la primalité d'un nombre a, il suffit qu'il ne soit divisible par aucun en entier entre 2 et $\sqrt{a}$.

3 On choisit $p = 5$ et $q = 53$. Vérifier de manière certaine que ces deux nombres sont premiers.

4 Parmi les valeurs suivantes $\{12, 91, 125, 143, 169\}$, choisir l'unique valeur de la clef privée a compatible avec les conditions requises pour son choix. Pour les autres valeurs, on indiquera pourquoi ce choix est inadéquat.

5 En déduire la valeur de clef privée b.

6 Quelles sont les information minimales qui doivent être transmise à un correspondant afin de lui permettre d'effectuer un chiffrement RSA ?

7 Combien de bits peuvent être transmis par un bloc RSA défini ainsi ? Ce nombre de bits sera calculé comme le plus grand nombre de bits possible pouvant être porté par n.

8 Les messages à transmettre sont codés avec l'alphabet $(\mathrm{e}, \mathrm{r}, \mathrm{s}, \mathrm{u})$. De combien de blocs RSA ai-je besoin pour envoyer le message "serrures" ? Donner alors la valeur des blocs correspondant au texte clair en utilisant un codage à taille fixe pour chacune des lettres.

9 Chiffrer les deux premiers bloc du message. Si $b > 10$, on utilisera l'algorithme d'exponentiation modulaire.

10 Quel est l'intérêt d'utiliser une hasardisation du message ?

11 Pour l'hasardisation, on veut utiliser les fonctions binaires d'expansion et de condensation suivantes :

- expansion $G : \mathbb{B}^3 \to \mathbb{B}^5$ définie comme $G(b_0b_1b_2) = b_0b_1b_2b_1b_0$.
- condensation $H : \mathbb{B}^5 \to \mathbb{B}^3$ définie comme $H(b_0b_1b_2b_3b_4) = b_0b_1b_2 \oplus b_2b_3b_4$.

En déduire combien de bits seront utilisés par bloc pour la clef aléatoire et le message.

12 Montrer, en hasardisant uniquement le premier bloc de message, qu'en changeant la clef aléatoire, on produit des messages hasardisés différents à partir du même premier bloc du message.

13 Déchiffrer les deux blocs du message hasardisés $\left[\begin{array}{cc} 173 & 253 \end{array}\right]$. Si $a > 10$, on utilisera l'algorithme d'exponentiation modulaire. On ne calculera à cette question que l'exponentiation modulaire.

14 A partir des résultats des blocs obtenus à la question précédente, déshasardisez chaque bloc.

15 En déduire le message codé transmis dans ces deux blocs.

16 Utiliser la méthode rapide d'exponentiation (= la méthode utilisant les restes chinois) pour déchiffrer le premier bloc du message.

Exercice 6: Tests de primalité

1 Soit un nombre entier stocké sur n bits. On suppose que le bit de poids le plus fort soit à 1. De combien de chiffres a-t-on besoin pour représenter ce nombre en base 10 ?

2 Appliquer la formule obtenue à la question précédente afin de déterminer le nombre p de chiffres a un nombre entier stocké sur 512 bits.

3 En déduire la borne inférieure du nombre de nombres premiers qui ont exactement 512 chiffres.

4 Tester la primalité de 1729 en effectuant deux fois le test de Fermat (*i.e.* $k = 2$).

5 Tester la primalité de 1729 avec le test de Miller-Rabin.

6 Pour les tests de primalité effectué aux deux questions précédentes, peut-on dire quelque chose sur la chance effective que 1729 soit effectivement premier dans chacun des cas ?

7 Ce résultat était-il prévisible ?

Exercice 7: Gain et information

A la suite d'un meurtre lors d'une réception dans un lieu clos, une liste de n suspects $(s_1, \ldots, s_n)$ est établie sur la base des personnes présentes. On note X la loi de probabilité associée ($\Pr[X = s_i]$ est la probabilité pour s_i d'être le meurtrier).

1. Dans un premier temps, nous n'avons aucune information sur qui peut être le meurtrier. Quelle est la probabilité pour chacun des suspects d'être le meurtrier ?
2. Calculer alors l'entropie de X.
3. A la suite d'une information, la police est presque sûre que le suspect s_1 est l'assassin, pendant que l'on a toujours aucune information sur les autres suspects. On a alors $\Pr[X = s_1] = 1 - \epsilon$.
4. Calculer alors l'entropie de X avec cette information supplémentaire.
5. A la suite d'une nouvelle information, s_1 est complètement écarté de la liste des suspects. Calculer alors les nouvelles probabilités de la loi X.
6. Calculer alors l'entropie de X avec cette information supplémentaire.
7. Que peut-on alors toujours en déduire sur l'ajout d'information et l'entropie ?

Exercice 8: Compression

Soit $\Sigma = (\mathtt{a}, \mathtt{b}, \mathtt{c})$ l'alphabet d'une source S. Cette source produit la sortie S suivant :

```
acbb bbbb bbcb bbbb abbc bbbb bbbb cbbb abbb bcbb bbbb bbcb
abbb bbbc bbbb bbbb cbbb bbbb bcbb bbbb abcb bbbb bbbc bbbb
abbb cbbb bbbb bcbb abbb bbcb bbbb bbbc abbb bbbb cbbb bbbb
```

Les espaces ne font pas partie du message à coder mais sont là pour améliorer sa lisibilité.

(1) **Codage entropique**
   (a) Donner les probabilités d'apparition des symboles pour la source S.
   (b) Calculer l'entropie de la source S.
   (c) Calculer alors la taille en bits qu'aurait le message S si l'entropie était atteinte.

(2) **Codage LZ78 :** on veut maintenant effectuer le codage avec la méthode LZ78 **avec une taille maximale de dictionnaire de 16** (il sera compté 0 à l'exercice si cette contrainte n'est pas prise en compte).
   (a) Effectuer le codage `LZ78` sur la source.
   (b) Donner la taille du codage `LZ78` produit. Le codage de la sortie devra être optimisé en faisant en sorte que le codage de l'indice dans le dictionnaire dépendant de la taille du dictionnaire au moment du codage (*i.e.* comme vu en TD).

(3) **Codage PPM d'ordre 2 :** après la lecture des 136 premiers caractères, on a généré l'ensemble des contextes d'ordre 2 suivants :
   - **ordre 0 :** (3/8/113/15).
   - **ordre 1 :** $a = (2/0/7/1)$, $b = (3/6/92/14)$, $c = (2/1/14/0)$.
   - **ordre 2 :** $\{ac, ca\} = (1/0/1/0)$, $\{cb, bc\} = (2/1/13/0)$, $\{ab, ba\} = (2/0/6/1)$, $\{cb, bc\} = (2/1/13/0)$, $bb = (3/5/73/13)$.

   où l'écriture $(2/0/6/1) = (n_\Delta, n_a, n_b, n_c)$ indique le comptage n_x du caractère x, et $\{ac, ca\} = (1/0/1/0)$ signifie que le comptage $(1/0/1/0)$ a été rencontré dans les contextes ac et ca.
   (a) Appliquer l'algorithme PPM d'ordre 2 des 8 derniers caractères du texte en utilisant les ordres et les contextes donnés ci-dessus. On écrira la totalité des mises à jour des contextes (seules les mise-à-jour seront données).
   (b) Quel est le nombre de bits engendré par le codage de ces derniers caractères ? On calculera explicitement la probabilité conditionnelles dans ces contexte.
   (c) Que fait-on lorsque l'on effectue une codage PPM d'ordre 0 ?
   (d) En supposant que l'ensemble des contextes ont déjà été créé, et que les probabilités des différents contextes ne changent plus, quel est la taille codée avec PPM d'ordre 1.
   (e) Même question avec un codage PPM d'ordre 2.