

# **Exercices et problèmes de cryptographie**



**Damien Vergnaud**

Préface de Jacques Stern

# **Exercices et problèmes de cryptographie**



**2<sup>e</sup> édition**

DUNOD

Illustration de couverture :  
*Energy of fractal realms* © agsandrew-Fotolia.com

<p>Le pictogramme qui figure ci-contre mérite une explication. Son objet est d'alerter le lecteur sur la menace que représente pour l'avenir de l'écrit, particulièrement dans le domaine de l'édition technique et universitaire, le développement massif du photocopillage.</p> <p>Le Code de la propriété intellectuelle du 1<sup>er</sup> juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée dans les établissements</p>	<p>d'enseignement supérieur, provoquant une baisse brutale des achats de livres et de revues, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.</p> <p>Nous rappelons donc que toute reproduction, partielle ou totale, de la présente publication est interdite sans autorisation de l'auteur, de son éditeur ou du Centre français d'exploitation du droit de copie (CFC, 20, rue des Grands-Augustins, 75006 Paris).</p>
--	--



© Dunod, 2012, 2015

5 rue Laromiguière, 75005 Paris  
www.dunod.com

ISBN 978-2-10-072145-0

Le Code de la propriété intellectuelle n'autorisant, aux termes de l'article L. 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

# PRÉFACE

« Pour devenir habile en quelque profession que ce soit, il faut le concours de la nature, de l'étude et de l'exercice ». Cette maxime d'Aristote semble bien mal s'appliquer à la cryptologie tant l'exercice y est absent. Il existe de multiples ouvrages de référence de qualité mais, pour la plupart, ils sollicitent très peu l'initiative des étudiants. Et même ceux – rares – qui sont accompagnés d'un véritable choix de problèmes à résoudre, par exemple sous forme d'un livre compagnon, ne couvrent pas totalement une discipline qui connaît une évolution rapide. C'est donc un réel manque que vient combler le recueil que propose Damien Vergnaud.

Le livre que j'ai le plaisir de présenter est issu d'un vrai travail de terrain puisqu'il est le résultat de plusieurs années d'enseignement de la cryptologie à l'École normale supérieure. À l'évidence, l'auteur a beaucoup de talent pour éveiller l'intérêt des étudiants et les conduire, pas à pas, à s'appropriier les concepts et les méthodes de la science du secret. Beaucoup de culture également, puisque les sujets choisis sont extrêmement variés à l'image d'une science qui emprunte à l'algèbre, à la théorie des probabilités, à l'algorithmique, à la théorie de l'information. D'ailleurs, ils débordent largement le cadre strict de la cryptographie. Ce talent et cette culture conduisent à un choix d'exercices qui ne demandent pas simplement à l'étudiant de faire des gammes mais lui proposent de s'attaquer à de véritables compositions : ici un effort raisonnable de programmation illustre des cryptanalyses célèbres comme celle de l'Enigma ou celle du programme Venona qui a permis l'interception de communications où les services russes mettaient incorrectement en œuvre le chiffrement jetable ; là une invitation à « mettre la main à la pâte » permet d'entrer de plain-pied dans les méthodes modernes de cryptanalyse – différentielle et linéaire – des algorithmes conventionnels tels que le DES ou l'AES ; là encore, une initiation progressive aux méthodes de factorisation d'entiers, intimement liées à la sécurité du RSA est proposée.

Présenter un tel ouvrage comme un simple livre d'exercices est le reflet de la modestie de son auteur. Certes, il permet la pratique nécessaire à l'acquisition des éléments essentiels de la cryptologie. Mais il va au-delà de cet objectif : chaque chapitre inclut une présentation qui est un véritable cours d'introduction et l'ensemble constitue de fait une forme d'ouvrage d'enseignement avancé fondé sur la pratique. En d'autres termes, le lecteur qui va au terme de tous les exercices proposés est déjà un

## Exercices et problèmes de cryptographie

véritable spécialiste, capable de se confronter aux multiples concepts que la cryptologie moderne a développés ces trente dernières années. À un moment où la cryptologie est au cœur de la société de l'information, de l'internet aux moyens de paiement en passant par les téléphones portables, une telle expertise est indispensable et il faut souhaiter au livre de Damien Vergnaud des lecteurs à la fois nombreux et actifs.

Jacques STERN  
Professeur à l'École normale supérieure

# TABLE DES MATIÈRES

<b>Préface</b>	<b>V</b>
<b>Avant-propos</b>	<b>XIII</b>
<b>Notations</b>	<b>XV</b>
<b>Chapitre 1. Cryptographie classique</b>	<b>1</b>
1.1 Chiffrement par substitution mono-alphabétique	1
Exercice 1.1 (avec programmation). Chiffrement de César	3
Exercice 1.2 (avec programmation). Chiffrement affine	4
Exercice 1.3 (avec programmation).	
Chiffrement par substitution mono-alphabétique	5
1.2 Chiffrement par substitution poly-alphabétique	8
Exercice 1.4 (avec programmation).	
Chiffrement de Vigenère – test de Kasiski	9
Exercice 1.5 (avec programmation).	
Chiffrement de Vigenère – indice de coïncidence	11
Exercice 1.6. Chiffrement de Hill – nombre de clés	12
Exercice 1.7. Chiffrement de Hill – attaque à clair connu	13
1.3 Chiffrement par transposition	14
Exercice 1.8 (avec programmation). Scytale	15
Exercice 1.9 (avec programmation).	
Chiffrement par transposition par colonnes	16
1.4 Chiffrement parfait	17
Exercice 1.10. Carré latin	18
Exercice 1.11 (avec programmation).	
Mauvaise utilisation du chiffrement jetable	20
Problème 1.12. Algorithme de Viterbi	20
1.5 La machine Enigma	22
Exercice 1.13. Enigma – Nombre de clés	24
Exercice 1.14 (avec programmation). Enigma – Tableau de connexions	25
Problème 1.15. Enigma – Indice de coïncidence	27
<b>Chapitre 2. Chiffrement par bloc</b>	<b>31</b>
2.1 Modes opératoires	32
Exercice 2.1. Modes opératoires et propriétés de sécurité	34
Exercice 2.2. Mode opératoire CBC *	36
Problème 2.3. Attaque sur le mode CBC avec le processus de bourrage RFC2040	38
2.2 Schémas de Feistel	39
Exercice 2.4. Schéma de FEISTEL à un ou deux tours	40
Exercice 2.5. Sécurité du schéma de FEISTEL à trois tours	42
Exercice 2.6. Distingueur pour le schéma de FEISTEL à trois tours*	43

## Exercices et problèmes de cryptographie

2.3	Chiffrement DES	45
	Exercice 2.7. Clés faibles et semi-faibles du chiffrement DES	46
	Exercice 2.8. Propriété de complémentation du chiffrement DES	48
	Exercice 2.9. Chiffrement DES avec blanchiment	49
	Exercice 2.10. Construction de Even-Mansour	50
	Exercice 2.11. Double chiffrement	51
	Exercice 2.12. Chiffrement Triple-DES avec deux clés indépendantes	52
	Exercice 2.13. Mode opératoire CBC-CBC-ECB	53
2.4	Chiffrement AES	55
	Exercice 2.14 (avec programmation). S-Boîte de l'AES	57
	Exercice 2.15 (avec programmation). Opération MixColumns	59
	Exercice 2.16. Propriétés de l'opération MixColumns	61
	Exercice 2.17 (avec programmation). Diversification de clé de l'AES	63
<b>Chapitre 3. Fonctions de hachage – Techniques avancées de cryptanalyse</b>		<b>65</b>
3.1	Généralités sur les fonctions de hachage	66
	Exercice 3.1. Résistance à la pré-image et aux collisions	66
	Exercice 3.2. Construction de Merkle-Damgård	67
	Exercice 3.3 (avec programmation). Collisions sur la fonction MD5 tronquée	71
3.2	Chiffrement par bloc et fonction de compression	72
	Exercice 3.4. Chiffrement par bloc et fonction de compression	72
	Exercice 3.5. Sécurité de la construction de Matyas-Meyer-Oseas avec le DES	73
	Exercice 3.6. Attaque en pré-image pour la construction de M. O. RABIN	74
3.3	Attaques génériques sur les fonctions de hachage itérées	76
	Exercice 3.7. Multi-collisions pour les fonctions de hachage itérées	76
	Exercice 3.8. Attaque en collision contre fonctions de hachage concaténées	78
	Problème 3.9. Attaque de Kelsey-Schneier	79
3.4	Cryptanalyse différentielle	82
	Exercice 3.10 (avec programmation). Table des différences du DES	83
	Problème 3.11. Cryptanalyse différentielle de FEAL-4	85
3.5	Cryptanalyse différentielle impossible	89
	Exercice 3.12. Attaque par différentielle impossible contre DEAL	89
	Problème 3.13. Attaque par différentielle impossible contre l'AES	92
3.6	Cryptanalyse linéaire	96
	Exercice 3.14 (avec programmation). Table d'approximation linéaire du DES	96
	Exercice 3.15. Approximation linéaire de l'addition	98
	Problème 3.16. Cryptanalyse linéaire de SAFER	100
	Exercice 3.17. Biais de la parité d'une permutation	102
3.7	Attaques par saturation	104
	Problème 3.18. Attaque par saturation contre l'AES	104
	Exercice 3.19. Attaque par distingueur sur Ladder-DES	108

<b>Chapitre 4. Chiffrement par flot</b>	<b>111</b>
4.1 Registres à décalage à rétroaction linéaire	111
Exercice 4.1. LFSR et polynômes de rétroaction	113
Exercice 4.2. Propriétés statistiques d'une suite produite par un LFSR	115
Exercice 4.3. Reconstruction du polynôme de rétroaction minimal	115
4.2 Chiffrement par flot par registres à décalage irrégulier	116
Exercice 4.4 (avec programmation). Distingueur sur le générateur à signal d'arrêt	117
Problème 4.5. Propriétés du générateur par auto-rétrécissement	119
4.3 Chiffrement par flot par registre filtré	120
Exercice 4.6. Attaque « deviner et déterminer » sur Toyocrypt	121
Exercice 4.7. Attaque algébrique sur Toyocrypt*	122
4.4 Chiffrement par flot par registres combinés	124
Exercice 4.8. Attaque par corrélation sur le générateur de Geffe	125
Exercice 4.9. Attaque « deviner et déterminer » sur le générateur de Geffe	127
Exercice 4.10. Attaque algébrique sur le générateur de Geffe	127
4.5 Le chiffrement par flot A5/1	128
Exercice 4.11. États internes de A5/1	129
Exercice 4.12. Attaque par compromis temps-mémoire sur A5/1	131
Problème 4.13. Attaque « deviner et déterminer » sur A5/1	132
4.6 Le chiffrement par flot RC4	135
Exercice 4.14. Cryptanalyse de RC4 sans opération d'échange*	136
Exercice 4.15. Biais de la suite chiffrante produite par RC4	137
Problème 4.16. Attaque par recouvrement de clé sur RC4	139
<b>Chapitre 5. Problème du logarithme discret</b>	<b>143</b>
5.1 Logarithme discret dans un groupe générique	143
Exercice 5.1. Multi-exponentiation	145
Exercice 5.2. Algorithme de Shanks	146
Exercice 5.3. Algorithme $\rho$ de Pollard	148
Exercice 5.4. Algorithme de Pohlig-Hellman	151
Exercice 5.5 (avec programmation). Application de l'algorithme de Pohlig-Hellman	153
5.2 Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^*$	154
Exercice 5.6. Entiers friables	155
Problème 5.7. Méthode de Kraitchik – Calcul d'indice *	157
5.3 Problèmes algorithmiques liés au logarithme discret	161
Exercice 5.8. Auto-réductibilité du problème du logarithme discret	161
Exercice 5.9. Algorithme $\lambda$ de Pollard	163
Problème 5.10. Logarithme discret de petit poids de Hamming	166
5.4 Interpolation polynomiale de logarithme discret	169
Exercice 5.11. Polynôme d'interpolation du logarithme discret	169
Exercice 5.12. Interpolation polynomiale de logarithme discret – Borne inférieure	170

## Exercices et problèmes de cryptographie

<b>Chapitre 6. Factorisation des entiers et primalité</b>	<b>173</b>
6.1 Tests de primalité	173
Exercice 6.1. Certificats de primalité de Pratt	174
Exercice 6.2. Nombres pseudo-premiers de Fermat en base $a$	175
Problème 6.3. Nombres de Carmichael – Critère de Korselt	176
Exercice 6.4 (avec programmation). Recherche de nombres de Carmichael	178
Exercice 6.5. Test de primalité de Solovay-Strassen	181
Problème 6.6. Test de primalité de Miller-Rabin	183
Exercice 6.7. Identité de Agrawal-Kayal-Saxena	185
Exercice 6.8. Nombres de Fermat et test de primalité de Pépin	186
6.2 Méthodes exponentielles de factorisation	188
Exercice 6.9 (avec programmation). Factorisation par divisions successives	188
Exercice 6.10 (avec programmation). Factorisation par la méthode Fermat	189
Exercice 6.11. Algorithme de Lehman *	189
Exercice 6.12. Méthode $p - 1$ de Pollard	192
Exercice 6.13 (avec programmation). Factorisation par la méthode $p - 1$ de Pollard	193
Exercice 6.14. Algorithme $\rho$ de Pollard	194
6.3 Multi-évaluation de polynômes et algorithme de Pollard-Strassen	195
Exercice 6.15. Division euclidienne rapide par la méthode de Newton	196
Exercice 6.16. Multi-évaluation d'un polynôme univarié	198
Exercice 6.17. Algorithme de Pollard-Strassen	200
6.4 Racine carrée modulaire et factorisation	202
Exercice 6.18. Extraction de racine carrée modulo $p$	202
Exercice 6.19. Extraction de racine carrée modulo $p^l$	204
Exercice 6.20. Extraction de racine carrée modulo $N$	205
Problème 6.21. Carrés modulaires friables	207
Exercice 6.22. Factorisation et logarithme discret	211
<b>Chapitre 7. Chiffrement à clé publique</b>	<b>213</b>
7.1 Fonction RSA	213
Exercice 7.1. Fonction RSA et factorisation	214
Exercice 7.2. Auto-réducibilité du problème RSA	215
Problème 7.3. Sécurité des bits de la fonction RSA	217
7.2 Chiffrement RSA	219
Exercice 7.4. Sécurité du protocole de chiffrement RSA naïf	220
Exercice 7.5. RSA avec module commun	220
Exercice 7.6 (avec programmation). Diffusion de données chiffrées avec RSA	221
Exercice 7.7. Attaque de Wiener	223
Exercice 7.8 (avec programmation). Attaque de Wiener	224
Exercice 7.9 (avec programmation). RSA et clairs liés	226
Exercice 7.10. RSA et petits textes clairs	227
Problème 7.11. Implantation du chiffrement RSA et théorème chinois des restes	228
7.3 Mise en accord de clé de Diffie-Hellman	230
Exercice 7.12. Attaque par le milieu	231
Problème 7.13. Logarithme discret et Diffie-Hellman *	232

7.4	Chiffrement d'ElGamal et variantes	235
	Exercice 7.14. Sécurité du chiffrement d'ElGamal naïf	235
	Exercice 7.15. Sécurité des bits du logarithme discret	237
	Exercice 7.16. Attaque sur le chiffrement d'ElGamal par petit sous-groupe	239
<b>Chapitre 8. Signatures numériques</b>		<b>241</b>
8.1	Signatures basées sur la primitive RSA	241
	Exercice 8.1. Sécurité du protocole de signature RSA naïf	242
	Exercice 8.2. Sécurité des protocoles de signature de De Jonge et Chaum	243
	Exercice 8.3. Sécurité de $\mathcal{F}$ -RSA et propriétés de $\mathcal{F}$	245
	Exercice 8.4. Sécurité de $\mathcal{F}$ -RSA pour la recommandation CCITT	247
	Exercice 8.5. Sécurité de $\mathcal{F}$ -RSA avec encodage PKCS #1 v1.5	249
	Exercice 8.6. Contrefaçon existentielle de $\mathcal{F}$ -RSA avec redondance linéaire	252
	Exercice 8.7. Contrefaçon universelle de $\mathcal{F}$ -RSA avec redondance linéaire *	253
	Problème 8.8. Sécurité du protocole de signature de Boyd	254
8.2	Signatures d'ElGamal et variantes	258
	Exercice 8.9. Contrefaçon existentielle du schéma de signature d'ElGamal naïf	258
	Exercice 8.10. Contrefaçon universelle du schéma de signature d'ElGamal naïf	259
	Exercice 8.11. Vérification des signatures d'ElGamal	260
	Exercice 8.12. Fonction de hachage et sécurité des signatures de Schnorr	261
	Exercice 8.13 (avec programmation). Paramètres publics dans le protocole DSA	262
	Exercice 8.14. Clé temporaire et sécurité des signatures d'ElGamal	263
8.3	Signatures de Lamport et variantes	265
	Exercice 8.15. Sécurité et efficacité des signatures de Lamport	265
	Exercice 8.16. Espace de message de la signature de Lamport	266
	Exercice 8.17. Extension de l'espace des messages des signatures de Lamport *	267
	Exercice 8.18. Arbres de Merkle	269
	Problème 8.19. Sécurité du protocole de signature de Groth	271
<b>Bibliographie</b>		<b>275</b>
<b>Index</b>		<b>281</b>



# AVANT-PROPOS

La cryptologie est un ensemble de techniques permettant d'assurer la sécurité des systèmes d'information. Cette discipline permet notamment de conserver aux données leur caractère de confidentialité, de contrôler leur accès ou d'authentifier des documents. L'utilisation de la cryptographie est de plus en plus répandue et les utilisateurs des systèmes cryptographiques doivent être en mesure non seulement de comprendre leur fonctionnement mais aussi d'en estimer la sécurité.

Cet ouvrage s'adresse aux étudiants de second cycle d'informatique ou de mathématiques. Il s'est développé à partir de textes de travaux dirigés et de travaux pratiques proposés à des étudiants du Master parisien de recherche en informatique (MPRI) et aux élèves de première année de l'École normale supérieure. Il a été conçu pour aider à assimiler les connaissances d'un cours d'introduction à la cryptologie et à se préparer aux examens. Il présente les outils mathématiques et algorithmiques utiles en cryptographie et les fonctionnalités cryptographiques de base dans le cadre de la cryptographie symétrique et asymétrique.

Les exercices destinés aux étudiants de « master 1 » pourront cependant être abordés par un étudiant motivé de licence ayant un goût pour l'algorithmique dans ses aspects mathématiques et pratiques. À l'intention des étudiants plus avancés, nous avons inclus des énoncés plus difficiles qui sont alors signalés par un astérisque. Enfin, l'ouvrage sera utile aux enseignants de cryptologie qui y trouveront un support pour leurs travaux dirigés.

La cryptologie est liée à d'autres disciplines mathématiques et informatiques comme l'arithmétique, l'algèbre, l'algorithmique, ou la théorie de la complexité. Le bagage informatique et mathématique requis pour aborder ce livre est celui que l'on acquiert lors des deux premières années de licence ou en classes préparatoires scientifiques augmenté de quelques notions de théorie des nombres de niveau troisième année. Ces notions plus avancées font l'objet de brefs rappels qui n'ont cependant pas pour ambition de remplacer un livre de cours.

Le but de cet ouvrage est de permettre à ceux qui le souhaitent de s'initier à la cryptographie par l'exemple. Il propose plus d'une centaine d'exercices et problèmes entièrement utilisés dans le cadre de travaux dirigés, de travaux pratiques ou d'examens. Ces exercices sont entièrement corrigés mais le lecteur ne tirera profit de ce livre que s'il cherche des solutions personnelles avant d'en étudier les corrections. L'étude de la cryptologie moderne ne peut se concevoir sans un ordinateur à portée

## Exercices et problèmes de cryptographie

de main et le livre propose de nombreux exercices de programmation qui ont pour but notamment d'acquérir une pratique de la cryptanalyse. Les données numériques de ces exercices sont disponibles en ligne sur :

[www.dunod.com/contenus-complementaires/9782100721450](http://www.dunod.com/contenus-complementaires/9782100721450)

Le lecteur pourra recopier les énoncés des exercices avant de les traiter. Il trouvera également une vingtaine d'exercices supplémentaires et des références complémentaires.

Cette **deuxième édition** a été inspirée par les nombreuses demandes et remarques que m'ont envoyées des étudiants et collègues, utilisateurs de la première édition. Cette nouvelle édition m'a donné l'occasion de modifier et réécrire des parties importantes du texte en suivant ces remarques sur le contenu, le style et l'organisation de l'ouvrage. En plus d'épurer le texte de ses inévitables erreurs typographiques et coquilles, j'ai simplifié et clarifié une grande partie des énoncés des exercices et de leurs solutions. J'ai notamment ajouté des questions intermédiaires pour simplifier la résolution de certains exercices et détaillé certains points techniques dans des solutions d'exercices complexes. J'ai supprimé certains exercices jugés trop difficiles et j'en ai également ajouté de nouveaux. Enfin, les compléments en ligne qui accompagnent l'ouvrage ont été enrichis d'une vingtaine d'exercices supplémentaires (avec leurs solutions complètes) et d'autres exercices et compléments de cours seront ajoutés progressivement.

## Remerciements

J'adresse un chaleureux merci à DAVID NACCACHE et JACQUES STERN avec qui j'ai eu le plaisir d'enseigner le cours d'*Introduction à la cryptologie*. Ma gratitude va également aux étudiants du MPRI et aux élèves normaliens de ces dernières années qui ont testé, malgré eux, la majorité des exercices présentés dans ce livre. Ce texte doit beaucoup à des conversations de couloirs et je tiens également à remercier les doctorants, post-doctorants et membres permanents de l'équipe Cryptographie de l'ENS – et particulièrement PIERRE-ALAIN FOUQUE – pour toutes les discussions que nous avons pu avoir. Pour terminer, je voudrais remercier AURÉLIE BAUER, GUILHEM CASTAGNOS, CÉLINE CHEVALIER, AURORE GUILLEVIC, PIERRE-ALAIN FOUQUE, FABIEN LAGUILLAUMIE, ROCH LESCUYER et JULIETTE VERGNAUD-GAUDUCHON pour la rigueur et la pertinence de leurs nombreux commentaires.

# NOTATIONS

Les conventions et notations suivantes sont utilisées dans cet ouvrage :

## a) Ensembles

Nous utilisons les notations ensemblistes classiques :  $\emptyset$  désigne l'ensemble vide ;  $x \in A$  signifie que  $x$  est un élément de l'ensemble  $A$ . Pour deux ensembles  $A$  et  $B$ ,  $A \subseteq B$  indique que  $A$  est un sous-ensemble de  $B$  (alors que  $A \subset B$  indique que  $A$  est un sous-ensemble strict de  $B$ ). De plus,  $A \cup B$  désigne la réunion de  $A$  et  $B$ ,  $A \cap B$  désigne l'intersection de  $A$  et  $B$ ,  $A \setminus B$  l'ensemble des éléments de  $A$  qui ne sont pas dans  $B$  et  $A \times B$  le produit cartésien des ensembles  $A$  et  $B$ . Le cardinal d'un ensemble  $A$  est noté  $\#A$ . Nous utilisons les notations classiques suivantes pour désigner certains ensembles :

$\mathbb{N}$	ensemble des entiers naturels
$\mathbb{P}$	ensemble des nombres premiers
$\mathbb{Z}$	anneau des entiers relatifs
$\mathbb{Q}$	corps des nombres rationnels
$\mathbb{R}$	corps des nombres réels
$\mathbb{C}$	corps des nombres complexes
$(\mathbb{Z}/N\mathbb{Z})$	anneau des résidus modulo un entier $N \geq 1$
$\mathbb{F}_q$	corps fini à $q$ éléments
$\mathfrak{S}_A$	groupe de permutations de l'ensemble $A$
$A^*$	groupe des éléments inversibles d'un anneau $A$
$\mathcal{M}_\ell(A)$	anneau des matrices carrées $\ell \times \ell$ à coefficients dans un anneau $A$
$A[X]$	anneau des polynômes à une indéterminée $X$ à coefficients dans un anneau $A$

La lettre  $p$  désigne le plus souvent un nombre premier  $p \in \mathbb{P}$  et nous notons  $(p_n)_{n \geq 1}$  la suite croissante des nombres premiers (avec  $p_1 = 2, p_2 = 3, \dots$ ). Pour un polynôme  $P \in A[X]$ , nous notons  $\deg P$  le degré de  $P$ . La notation  $\mathbb{G}$  désigne un groupe dont la loi est notée multiplicativement. L'élément neutre pour la multiplication dans  $\mathbb{G}$  est noté  $1_{\mathbb{G}}$ . L'ordre d'un groupe  $\mathbb{G}$  est noté  $|\mathbb{G}| = \#\mathbb{G}$  et  $\langle g \rangle$  désigne le sous-groupe de  $\mathbb{G}$  engendré par  $g \in \mathbb{G}$ .

## b) Fonctions

Nous notons  $f : A \longrightarrow B$  pour indiquer que  $f$  est une fonction d'un ensemble  $A$  dans un ensemble  $B$ . Pour un sous-ensemble  $A' \subseteq A$ , nous notons  $f(A') = \{f(a), a \in A'\} \subseteq$

## Exercices et problèmes de cryptographie

$B$ . Pour un sous-ensemble  $B' \subseteq B$ , nous notons  $f^{-1}(B') = \{a \in A, f(a) \in B'\} \subseteq A$ . La composition de fonctions est notée  $\circ$ . Nous utilisons les notations classiques suivantes pour désigner certaines fonctions :

$\lfloor x \rfloor$	partie entière par défaut de $x \in \mathbb{R}$ ( $\lfloor x \rfloor \leq x < \lfloor x \rfloor + 1$ )
$\lceil x \rceil$	partie entière par excès de $x \in \mathbb{R}$ ( $\lceil x \rceil - 1 < x \leq \lceil x \rceil$ )
$\ln(x)$	logarithme népérien de $x \in \mathbb{R}$ ( $x > 0$ )
$\log(x)$	logarithme en base 2 de $x \in \mathbb{R}$ ( $x > 0$ )
$\log_g(h)$	logarithme discret de $h \in \langle g \rangle$ en base $g \in \mathbb{G}$
$\pi(x)$	nombre de nombres premiers inférieurs ou égaux à $x$ ( $\#\{p \in \mathbb{P}, p \leq x\}$ )
$\Psi(x, y)$	fonction de Dickman-De Bruijn ( $\#\{n \in \mathbb{N}, n \leq x \text{ et } n \text{ est } y\text{-friable}\}$ )
$\binom{n}{m}$	coefficient binomial $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ pour $0 \leq m \leq n$
$\left(\frac{x}{m}\right)$	symbole de Jacobi
$\text{pgcd}(a, b)$	plus grand commun diviseur de $a, b \in \mathbb{Z}$
$\text{ppcm}(a, b)$	plus petit commun multiple de $a, b \in \mathbb{Z}$
$\text{Pr}(E)$	probabilité d'un événement $E$

### c) Chaînes binaires

Nous utilisons les notations classiques suivantes sur les chaînes binaires :

$\{0, 1\}^n$	ensemble des chaînes binaires de longueur $n$
$\{0, 1\}^*$	ensemble des chaînes binaires de longueur finie
$\wedge$	ET logique (bit à bit pour deux chaînes de même longueur)
$\vee$	OU logique (bit à bit pour deux chaînes de même longueur)
$\neg$	NON logique (bit à bit pour deux chaînes de même longueur)
$\oplus$	« OU exclusif » (bit à bit pour deux chaînes de même longueur)
$ x $	longueur binaire d'une chaîne $x \in \{0, 1\}^*$
$\bar{x}$	chaîne binaire complémentaire de $x$ ( $\bar{x} = \neg x = x \oplus 1^n$ avec $n =  x $ )
$x  y$	concaténation des chaînes $x$ et $y$
$x^n$	concaténation de la chaîne $x$ $n$ fois $\underbrace{(x  \dots  x)}_{n\text{fois}}$
$\lll i$	rotation à gauche d'une chaîne de bits de $i$ positions

Dans les chapitres 2, 3 et 4, nous utilisons une fonte de type « machine à écrire » pour représenter la valeur d'un octet avec deux chiffres hexadécimaux :  $00 = 0$ ,  $01 = 1$ , ...,  $0A = 10$ , ...,  $10 = 16$ , ...,  $FF = 255$ .

#### d) Notations algorithmiques

Les algorithmes sont présentés sous forme de pseudo-code simple (notamment en s'affranchissant des problèmes de mémoire). Les entrées et les sorties sont toujours précisées. Les structures de contrôle classiques sont notées en gras (**tant que** condition **faire** instructions **fin tant que**, **si** condition **alors** instructions **sinon** instructions **fin si**, ...). Les commentaires dans les algorithmes sont signalés par le symbole  $\triangleright$ . Le symbole  $a \leftarrow b$  indique l'assignation algorithmique (*i.e.*  $a$  prend la valeur de  $b$ ) et le symbole  $a \xleftarrow{u.a.} A$  l'assignation d'un élément tiré uniformément aléatoirement (*i.e.* un élément est tiré uniformément aléatoirement dans l'ensemble  $A$  et la valeur obtenue est enregistrée dans  $a$ ).



# CRYPTOGRAPHIE CLASSIQUE

# 1

La cryptologie est une science très ancienne : les hommes ont toujours eu besoin de dissimuler des informations et de transmettre des messages en toute confidentialité. Le terme cryptologie vient du grec *kruptos* signifiant *secret, caché* et de *logos* signifiant *discours*. La cryptologie est donc la *science du secret*. Elle regroupe la cryptographie et la cryptanalyse : la première a pour but de concevoir des systèmes visant à assurer la sécurité des communications sur un canal public et la seconde vise à trouver des failles dans ces systèmes.

La cryptographie est traditionnellement utilisée pour dissimuler des messages aux yeux de certains utilisateurs et le chiffrement des communications militaires a depuis l'Antiquité été une préoccupation majeure des diverses forces armées. Le *chiffrement* regroupe les techniques mises en œuvre pour brouiller la signification d'un message qui est matériellement visible. Le contenu du message ne doit alors être retrouvé que par les personnes auxquelles le message est adressé. Le chiffrement fait appel à deux processus élémentaires impliquant la transformation des lettres d'un message pour satisfaire ces propriétés : la *substitution* qui consiste à remplacer, sans en bouleverser l'ordre, les symboles d'un texte clair par d'autres symboles et la *transposition* qui repose sur le bouleversement de l'ordre des symboles (mais pas leur identité).

Dans ce chapitre, nous allons étudier des systèmes de chiffrement relativement simples qui ont été utilisés de l'Antiquité (*chiffrement de César* ou *scytale*) jusqu'au début du XX<sup>e</sup> siècle (*chiffrement de Vernam, chiffrement de Hill, machine Enigma*).

## 1.1 CHIFFREMENT PAR SUBSTITUTION MONO-ALPHABÉTIQUE

Le *chiffrement par substitution* consiste à remplacer dans un message un ou plusieurs symboles par un ou plusieurs symboles (généralement du même alphabet) tout en conservant l'ordre de succession des symboles du message. Dans cette section, nous considérons le chiffrement par substitution *mono-alphabétique* qui consiste à remplacer chaque symbole individuel du message par un autre symbole de l'alphabet. Nous allons étudier les techniques de cryptanalyse permettant d'attaquer un tel système.

Elles reposent sur l'analyse des fréquences des symboles utilisés dans le texte chiffré et utilisent le fait que, dans chaque langue, certains symboles ou combinaisons de symboles apparaissent plus fréquemment que d'autres. Les systèmes de chiffrement par substitution mono-alphabétique conservent la répartition des fréquences et

si le message chiffré est suffisamment long, la recherche d'un symbole ayant une fréquence élevée permettra parfois de retrouver tout ou partie du message clair associé.

La fréquence d'apparition des lettres varie bien évidemment en fonction de la langue et du type de texte considérés. Pour un texte rédigé en français, nous obtenons généralement les fréquences d'apparition (en pourcentage) proches des valeurs suivantes<sup>1</sup> (cf. Figure 1.1) :

a	b	c	d	e	f	g	h	i	j	k	l	m
8,46	1,02	3,21	3,78	17,60	1,11	1,12	1,07	7,40	0,48	0	6,05	2,70
n	o	p	q	r	s	t	u	v	w	x	y	z
6,38	5,19	2,68	1,21	6,56	7,56	7,26	6,63	1,65	0	0,03	0,03	0,01

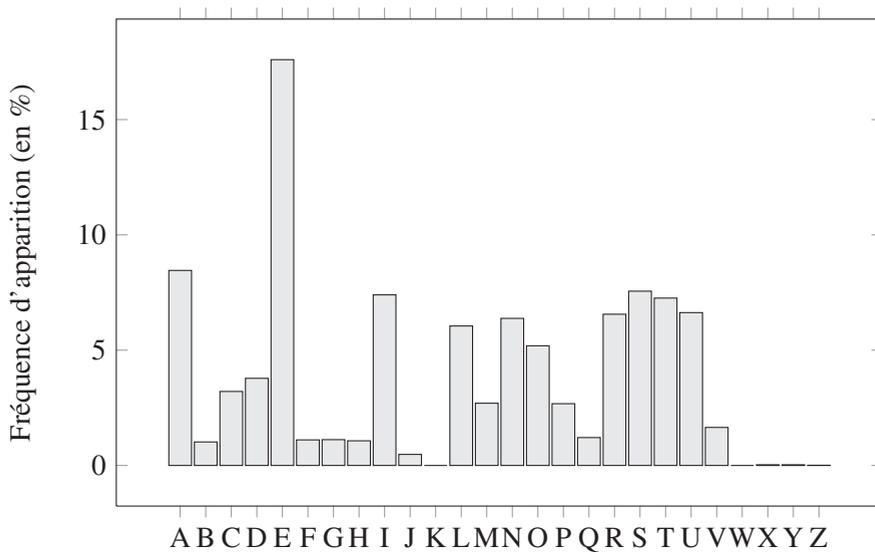


Figure 1.1- Fréquence d'apparition des lettres en français

De même, certains couples de lettres (ou *bigrammes*) apparaissent plus souvent que d'autres dans une langue donnée. Les vingt bigrammes les plus fréquents de la langue française sont (du plus fréquent au moins fréquent) : *es, de, le, en, re, nt, on, er, te, el, an, se, et, la, ai, it, me, ou, em* et *ie*.

Le chiffrement par substitution mono-alphabétique le plus simple est le *chiffrement par décalage*, aussi connu sous le nom de *chiffrement de César*. Il consiste simplement à décaler les lettres de l'alphabet d'un nombre de positions constant vers la

1. Ces valeurs correspondent aux fréquences d'apparition des lettres dans le roman *Notre-Dame de Paris* de V. Hugo (en identifiant les lettres accentuées et les lettres non accentuées).

## 1.1. Chiffrement par substitution mono-alphabétique

droite ou la gauche. Par exemple, en décalant les lettres de trois rangs vers la gauche (comme le faisait J. CÉSAR), le texte clair *veni vidi vici* devient *yhql ylgf ylff*.

### Exercice 1.1 (avec programmation). Chiffrement de César

Décrypter le texte suivant qui a été obtenu en appliquant le chiffrement de César sur un texte en langue française dans lequel les espaces ont été supprimées :

```
vcfgrwqwfsbhfsntowbsobgfsbhfsnqvsnjcigsgghqsoixcif  
rviwtshseicwbsgojsnjcigdogeisjcigoihfsgofhwgobjc  
igbsrsjsnqwfqizsfrobgzsgfiszgsgxcifgcijfopzsgioj  
sqzsggwubsgrsjchfsdfctsggwcbdozfzseiszsghhcbashwsf
```

### Solution

Le chiffrement de César est un mode de chiffrement par substitution, il ne modifie donc pas la fréquence d'apparition des lettres. La lettre la plus fréquente dans un texte français étant le « e », le décalage entre la lettre la plus fréquente dans ce texte chiffré et la lettre « e » doit donc nous révéler la clé utilisée pour le chiffrement. La lettre qui apparaît le plus souvent dans le texte chiffré est le « s » avec 33 occurrences (puis vient la lettre « g » avec seulement 24 occurrences). Le décalage utilisé est donc vraisemblablement de 14 lettres vers la gauche et l'on obtient le message clair suivant :

```
horsdicirentrezfaineansrentrezchezvousestceaujourd  
dhui fetequoinesavezvouspasquevousautresartisansvo  
usne devez circuler dans les rues les jours ouvrables qu  
e les signes de votre profession par le quel est ton metier
```

En ajoutant les espaces et la ponctuation, nous reconnaissons la première réplique de la pièce *Jules César* écrit par W. SHAKESPEARE en 1599 (dans la traduction de M. GUIZOT) :

*« Hors d'ici, rentrez, fainéans ; rentrez chez vous. Est-ce aujourd'hui fête ? Quoi ! Ne savez-vous pas que vous autres artisans vous ne devez circuler dans les rues les jours ouvrables qu'avec les signes de votre profession ? Parle, quel est ton métier ? »*

Le chiffrement affine est un système de chiffrement par substitution mono-alphabétique. La clé consiste en un couple d'entiers  $(a, b) \in (\mathbb{Z}/26\mathbb{Z})^* \times (\mathbb{Z}/26\mathbb{Z})$ . En remplaçant chaque lettre de l'alphabet par son rang (la lettre « a » est remplacée par 0, la lettre « b » est remplacée par 1, ... et la lettre « z » est remplacée par 25), une lettre du texte clair de rang  $i \in \{0, \dots, 25\}$  est remplacée dans le chiffré par la lettre de rang  $a \cdot i + b \pmod{26}$ . Puisque  $a$  est inversible dans  $(\mathbb{Z}/26\mathbb{Z})$ , cette transformation est bien une permutation de  $(\mathbb{Z}/26\mathbb{Z})$ .

**Exercice 1.2 (avec programmation).** Chiffrement affine

Décrypter le texte suivant qui a été obtenu en appliquant le chiffrement affine sur un texte en langue française dans lequel les espaces ont été supprimées :

ntjmpumgxpqtstgqpgtxpnchumtputgfsftgthnngxnchumwx  
 ootr tumhpyctgktjqtjchfooxujqhgztumxpotjxotfoqtohr  
 xumhzutwftgtopfmntjmpuatmfshodpfrxpjjtqtghbxuj

**Solution**

Comme le chiffrement de César, le chiffrement affine ne modifie pas la fréquence d'apparition des lettres. La lettre la plus fréquente dans un texte français est le « e » et les lettres suivantes sont par ordre de fréquence le « a », le « i », le « n », le « s » et le « t » (avec des fréquences très variables d'un texte à l'autre). La lettre qui apparaît le plus souvent dans le texte chiffré est le « t » avec 19 occurrences puis vient la lettre « m » avec 12 occurrences.

En supposons que la lettre « t » correspond à la lettre « e » et que la lettre « m » correspond à l'une des lettres « a », « i », « n », « s » ou « t », nous devons résoudre pour chaque choix un système linéaire à deux équations et deux inconnues dans  $(\mathbb{Z}/26\mathbb{Z})$  de la forme

$$\begin{cases} a \cdot 19 + b = 4 \\ a \cdot 12 + b = \ell \end{cases}$$

où  $\ell$  est l'entier 0, 8, 13, 18 ou 19 selon que la lettre testée pour « m » est « a », « i », « n », « s » ou « t » (respectivement). Le couple  $(a, b)$  obtenu est testé en déchiffrant les premiers caractères du chiffré et nous obtenons les résultats suivants :

Lettre testée	« a »	« i »	« n »	« s »	« t »
$\ell$	0	8	13	18	19
$(a, b)$	(8, 8)	(18, 0)	(21, 21)	(24, 16)	(9, 15)
Début du « clair » associé	ie caym	aegikw	iecn yz	qeysmc	cestun

La clé à utiliser pour le déchiffrement est donc vraisemblablement le couple (9, 15) et nous obtenons le message clair suivant :

cestunroudeverdureouchanteuneriviereaccrochantfol  
 lementauxherbesdeshailionsdargentoulesoleildelamon  
 tagnefiereluitcestunpetitvalquimoussederayons

Il s'agit bien sûr du premier quatrain du sonnet *Le Dormeur du val* écrit par A. RIMBAUD en 1870.

Les systèmes de chiffrement par substitution mono-alphabétique les plus généraux utilisent une permutation aléatoire des symboles de l'alphabet utilisé. Pour l'alphabet latin formé de 26 lettres, le nombre de clés possibles est donc égal à  $26! \approx 4 \cdot 10^{26} \approx 2^{88}$ . Cependant même si le nombre de clés rend toute recherche

## 1.1. Chiffrement par substitution mono-alphabétique

exhaustive impossible, les techniques d'analyse fréquentielle permettent de décrypter facilement un chiffré suffisamment long.

### **Exercice 1.3 (avec programmation).**

#### *Chiffrement par substitution mono-alphabétique*

Le texte suivant résulte du chiffrement d'un texte français par une substitution mono-alphabétique.

v ubcfb osu ymoqsuu n cxqfj dqmfnu ub vjcfqu juz amqjmrz zmscfsb bqflu  
auoqz hfszbms zwfba ju wusbms qusbqu ncsz ju vmo z uddmqvcfb n uxfbuq ju  
xusb wcoxcfz fj eczzc qcefnuwusb jc emqbu xfbqu no ijmv nuz wcfzmsz nu jc  
xfvbmfcu ecz czzul qcefnuwusb vueusncsb emoq uweuvauq kou z usrmoddqu us  
wuwu buwez kou jof os bmoqifjjms nu emozzfuqu ub nu zciju  
ju acjj zusbcfb ju vamo vofb ub ju xfuog bcefz c j osu nu zuz ugbquwfbuz  
osu cddfvau nu vmojuoq bqme xczbv emoq vu nuejmfuwusb fsbuqfuq ubcfb  
vjmoou co woq ujju quequzusbcbf zfwjuwusb os usmqwu xfzcru jqcru nu ejoz  
n os wubqu ju xfzcru n os amwwu n usxfqms kocqcsbu vfsk csz c j uecfzsu  
wmozbcvau smfcu cog bqcfbz cvvusbouz ub iucog

hfszbms zu nqfruc xuqz j uzvcjfuq fj ubcfb fsobfju n uzzcpuq nu  
eqsnqu j czvuszuq wuwu cox wufjuoquq uemkouz fj dmsvbfmsscfc qcuwusb  
cvboujjuwusb n cfjuoqz ju vmoqcsb ujvubqfkou ubcfb vmoeu ncsz jc ymoqsuu  
v ubcfb osu nuz wuzoquz n uvmsmfwu eqfzuz us xou nu jc zuwcfsu nu jc acfsu  
zms ceecqbuwusb ubcfb co zuebfuwu hfszbms kof cxcfb bqsbu suod csz ub  
zmoddqcfc n os ojvuqu xcqfkouog co nuzzoz nu jc vauxfju nqmfbu wmsbcfb  
jusbwusb fj z cqquc ejozfuqz dmfc us vauwfs emoq zu quemzuq c vackou  
ecjfuq zoq osu cddfvau vmjjuo co woq dcvu c jc vcru nu j czvuszuq j  
usmqwu xfzcru xmoz dfgcfb no qurcqn v ubcfb os nu vuz emqbqcfbz cqcsru  
nu bujju zmqbu kou juz puog zuwibus zofxqu vujof kof eczzu osu jurusnu  
zmoz ju emqbqcfb nfzcfb ifr iqmbauq xmoz qurcqn

c j fsbuqfuq nu j ceecqbuwusb nu hfszbms osu xmfg zovquu dcfzcfb usbusnu  
osu zuqfu nu smwiqz kof cxcfb bqcfb c jc eqmnovbms nu jc dmsbu jc xmfg  
eqmxuscfc n os ejkou nu wubcj mijmsrou wfqmfq buqsu usvczbqu ncsz ju woq  
nu nqmfbu hfszbms bmoqsc os imobms ub jc xmfg nfwfoc nu xmjowu wcfz juz  
wmbz ubcfbus usvmqu nfzbfsvbz ju zms nu j ceecqufj no bujuvqcs vmwwu ms  
nfzcfb emoxcfb ubqu czzmoqnf wcfz fj s p cxcfb covos wmpus nu j ubufsnqu  
vmwejubuwusb hfszbms zu nqfruc xuqz jc dusubqu fj ubcfb nu zbcboqu dquju  
ejomb eubfbu ub zc wfrquq ubcfb zmojfrsuu ecq jc vmwifscfzms ijuou  
osfdmqwu no ecqbf fj cxcfb juz vauuog bqz ijmsnz ju xfzcru scoqujjuwusb  
zcsrofs jc euco noqvfu ecq ju zcxms rqmzzfuq juz jcwuz nu qczmfc uwmozzuuz  
ub ju dqmfnu nu j afxuq kof xuscfc nu eqsnqu dfs

Décrypter ce texte.

**Solution**

Le nombre d'occurrences de chaque caractère du texte chiffré est donné dans le tableau suivant :

a	b	c	d	e	f	g	h	i	j	k	l	m
17	141	150	23	48	146	11	6	13	106	14	2	95
n	o	p	q	r	s	t	u	v	w	x	y	z
71	113	4	129	20	129	0	329	51	57	35	2	128

Il est vraisemblable que le caractère « u » représente le caractère « e » dans le texte clair (nous utiliserons une fonte grasse pour indiquer les lettres appartenant au texte clair). Les autres lettres les plus fréquentes sont le « c » et le « f » mais leurs fréquences sont trop proches pour décider quels caractères elles représentent.

Les bigrammes les plus fréquents de la langue française sont « es », « de », « le », « en », « re » « nt », et « on ». Les bigrammes commençant par « u » dans le texte chiffré ne sont pas assez fréquents pour décider quel caractère correspond à la lettre « s » dans le texte clair. Les bigrammes les plus fréquents du texte chiffré finissant par « u » sont « ju » (39 occurrences) et « nu » (35 occurrences) ce qui suggère que « j » représente « l » et « n » représente « d ». Ces trois substitutions donnent, pour le premier paragraphe, le texte suivant :

v ebcfb ose ymoqsee d cxqfl dqmfde eb vlcfqe lez amqlmrez zmsscfe**sb**  
 bqefle aeoqez hfszbms zwfba le wesbms qesbqe dcsz le vmo z eddmqvcfb  
 d exfbeg le xesb wcoxcfz fl eczzc qcefdewesb lc emqbe xfbqee do ilm**v**  
 dez wcfzmsz de lc xfvbmfqe ecz czzel qcefdewesb veeesdcsb emoq eweevaeq  
 koe z esrmoddqe es wewe bewez koe lof os bmoqifllms de emozzfeqe eb de  
 zcile

Le mot « do » suggère que le caractère « o » représente « u ». Avec cette déduction, le mot « ose » devient « use » qui suggère que le caractère « s » représente « n ». Le premier paragraphe du texte devient alors :

v ebcfb une ymuqnee d cxqfl dqmfde eb vlcfqe lez amqlmrez zmnn**cfenb**  
 bqefle aeuqez hfnzbmn zwfba le wenbmn qenbqe dcnz le vmu z eddmqvcfb  
 d exfbeg le xenb wcu**xcfz fl** eczzc qcefdewenb lc emqbe xfbqee du ilm**v**  
 dez wcfzmnz de lc xfvbmfqe ecz czzel qcefdewenb vee**endcnb** emuq eweevaeq  
kue z enr**muddqe en** wewe bewez kue luf un bmuqifllmn de emuzzfeqe eb de  
 zcile

Le mot « kue » qui apparaît deux fois suggère que la lettre « k » représente « q ». Les mots « qcefdewenb » et « eb » suggèrent que la lettre « b » représente « t ». Les mots « luf » et « fl » suggèrent que la lettre « f » représente « i » et nous obtenons :

v etcit une ymuqnee d cxqil dqmide et vlciqe lez amqlmrez zmnn**cient**  
 tqeile aeuqez hinztmn zwita le wentmn qentqe dcnz le vmu z eddmqvcit