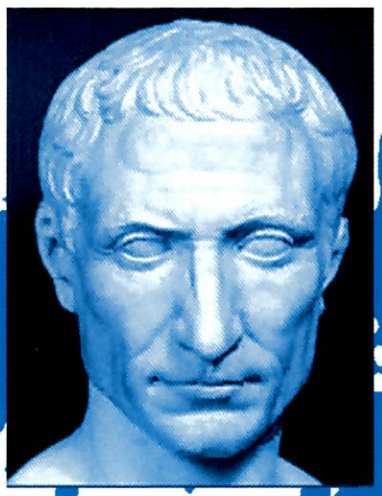
Math Jeunes



26ème année - N°110 S - Janvier 2005

Math-Jeunes et Math-Jeunes Junior

Revues trimestrielles publiées par la
Société Belge des Professeurs de Mathématique d'expression française

Secrétariat : M.-C. Carruana, S.B.P.M.e.f., rue de la Halle 15, 7000 Mons, ☎-🖷 32-(0)65-373729,
e-mail : sbpm@sbpm.be, URL : http://www.sbpm.be.

Trois numéros de chaque revue sont publiés annuellement. Pour connaître la liste des anciens numéros encore disponibles et leur prix selon votre localisation, adressez-vous au secrétariat.
Les abonnements destinés aux élèves de l'enseignement secondaire sont de préférence pris par l'intermédiaire d'un professeur. Effectuez vos paiements :

- ☞ pour la Belgique : par virement au Compte n°000-0728014-29 de *S.B.P.M.e.f.*, rue de la Halle 15, 7000 Mons
- ☞ pour la France : par virement au Compte CCP Lille 10 036 48 S
- ☞ pour les autres pays : par virement international au Compte IBAN BE26 0000 7280 1429, BIC BPOTBEB1 de la SBPMef. Les personnes résidant à l'étranger qui ne peuvent faire un virement à ce compte sont invitées à envoyer un mandat postal international. Tout chèque bancaire non encaissable en Belgique devra être majoré de 12,50 € pour frais d'encaissement.

Tarifs

Abonnements groupés (5 exemplaires au moins)				
	Une des deux revues		Les **deux** revues	
Belgique	4 €		7 €	
	✉	🖃	✉	🖃
Europe	7 €	9 €	13 €	17 €
Autres pays	10 €	14 €	15 €	20 €

Abonnements individuels				
	Une des deux revues		Les **deux** revues	
Belgique	6 €		11 €	
	✉	🖃	✉	🖃
Europe	13 €	16 €	16,50 €	20,50 €
Autres pays	15 €	21 €	20 €	25 €

Non prior : ✉, Prior : 🖃

Sommaire

En couverture : Julius Caius Caesar (voir « De la scytale à Enigma »), Ronald Rivest, Adi Shamir et Leonard Aldeman (voir « Le crypto-système RSA »).

Faites-nous savoir quel article, ou quelle rubrique, vous avez préféré(e). Envoyez un SMS au **0473-973808** ou un e-mail à **sbpm@sbpm.be**

Éditeurs responsables et rédacteurs en chef :
– pour *Math-Jeunes* : G. Noël, Rue de la Culée, 86, 6927 Resteigne
– pour *Math-Jeunes Junior* : A. Paternottre, Rue du Moulin 78, 7300 Boussu

N°110 Vingt-sixième année 2005

Math-Jeunes

Le codage

S'il est un thème qui est vaste, c'est bien celui du codage. Toute communication entre êtres vivants nécessite d'encoder une *information* en un *langage*. La langue maternelle constitue sans doute pour chacun d'entre nous sa première expérience d'encodage. Mais bien d'autres systèmes existent. Parfois, la nature choisit elle-même le codage utilisé. L'exemple le plus connu est sans doute celui du code ADN. *Math-Jeunes* a évoqué cette double hélice dans son numéro 107. Nous nous contenterons de la reproduire en filigrane sur cette page.

On dit souvent que la mathématique est elle-même un langage. Toute activité mathématique implique donc une ou plusieurs activités de codage. De plus, de nombreux registres de codage différents peuvent être utilisés. Du registre de la *langue naturelle*, au registre *symbolique*, en passant par divers registres *graphiques*, la même information peut être codée de façons variées. Souvent un registre s'avère plus efficace qu'un autre pour supporter le raisonnement.

On le voit, pour aborder toutes les questions relatives au codage, plusieurs dizaines de numéros de *Math-Jeunes* seraient nécessaires. Force nous est de limiter drastiquement notre propos. Aussi évoquera-t-on dans les pages qui suivent essentiellement deux thèmes : d'une part des codages « au quotidien », d'autre part des codages destinés à encrypter des messages secrets.

Dans la première catégorie, nous rencontrerons d'abord un article de Nadège Vandenabeele, qui analyse le codage utilisé dans un jeu assez répandu : le « Master Mind ». Denis Pierson évoque ensuite le problème de la *détection* des erreurs dans les numéros de comptes bancaires. Ce problème est repris plus loin par Pierre Dufour, qui montre la possibilité de *corriger* une erreur. Si l'on pense à la quantité d'informations véhiculées quotidiennement par les réseaux informatiques, on mesure l'importance de cette question. Dans son article, Pierre Dufour utilise de façon intensive la numération binaire. Ce codage des nombres, banal pour les informaticiens, est semblable au codage décimal, dont l'apprentissage constitue une des premières activités scolaires du jeune enfant. Aussi un article de Yolande Noël-Roch rappelle-t-il les principes fondamentaux de la numération de position.

Dans la seconde catégorie, figure d'abord un article de Julie Christophe et Julie De Saedeleer qui font un historique rapide du développement des systèmes cryptographiques. On ne s'étonnera guère de trouver à l'origine des motivations militaires. Aussi un des premiers personnages de cet historique est-il Jules César lui-même. La méthode des fréquences est également rappelée, (elle est utilisée dans la rubrique « Jeux »dans les jeux. La machine « Enigma » nous renvoie à la rubrique « Anniversaire » consacrée par Simone Trompler à Alan TURING. Mais de nos jours, l'espionnage économique dépasse peut-être en quantité l'espionnage militaire. Pour préserver le secret des transations financières, sans les entraver, de nouvelles méthodes permettent au rédacteur d'un message d'encrypter celui-ci dans un code que seul le destinataire peut décoder en un temps raisonnable. Ces techniques appelées « codages à clef révélée » sont décrites dans deux articles, dus à Isabelle Pays et Françoise Valette. Tous deux reposent sur des notions mathématiques relativement simples, les congruences de Gauss, auxquelles une page de rappel est consacrée.

ANNIVERSAIRE

Alan Turing

Simone Trompler

Alan Turing (1912–1954) est un mathématicien anglais. Ses recherches dans les relations entre les machines et la Nature ont créé le domaine de l'Intelligence Artificielle. Il a été un des premiers à entrer dans l'ère informatique.

Dès le début de sa scolarité, Turing montre d'excellentes aptitudes pour les mathématiques et les autres sciences. Par contre, l'histoire et les langues l'ennuient et ses résultats y sont médiocres. Il montre très vite une tendance à suivre ses propres idées et à négliger les indications données par les professeurs. Son éducation en mathématiques, très profonde, est due uniquement à ses lectures personnelles.

Il se passionne pour la course et deviendra un coureur remarquable, se servant de ce sport pour se détendre après un stress trop important.

A 19 ans, il entre à l'université de Cambridge avec une bourse et est diplômé à 22 ans ; ensuite, il suit des cours de mathématique avancée dans différents domaines, notamment en probabilités.

En 1936, il apporte une contribution importante à la théorie de la calculabilité en introduisant une machine abstraite (appelée maintenant machine de Turing) qui est censée établir la non-résolubilité de certains problèmes en passant d'une étape de calcul à la suivante par application d'un ensemble fini de règles.

Il part ensuite à l'université de Princeton et publie en 1939 un travail important *Systems of Logic Based on Ordinals.*

En 1938, il revient à Cambridge. Il y est contacté pour aider à déchiffrer les messages secrets allemands encodés à l'aide d'une machine appelée « Enigma([1]) ». La particularité de cette machine était de changer continuellement de code, ce qui rendait le décodage très difficile. Turing et ses collaborateurs y parvinrent en construisant une machine dénommée « Turing's Bombe ». Les concepts et techniques élaborés par Turing furent également exploités pour la construction d'une autre machine, la « Colossus » chargée de décrypter les messages échangés entre Hitler et ses généraux. Ceux-ci étaient encodés à l'aide d'une machine plus complexe encore que l'« Enigma ». La « Colossus » a été la première machine programmable et peut de ce fait être considérée comme l'ancêtre de nos ordinateurs.

Durant toute la guerre, Turing fut employé à plein temps, d'abord en Angleterre, puis aux USA, au décodage des messages des armées de l'air et de mer de l'ennemi. Il contribuera ainsi à sauver de très nombreuses vies.

Revenu en Angleterre, il travaille ensuite activement à la mise au point d'ordinateurs. Il pensait qu'une machine intelligente pourrait être construite sur le modèle du cerveau humain. En 1950 il écrit un article qui présente ce qu'on appelle actuellement le « test de Turing ». Dans ce test, une personne poserait des questions par l'intermédiaire d'un clavier à une autre personne et à la machine. Turing pense que, s'il est impossible de distinguer les réponses de la personne et celles de la machine, on peut parler de machine intelligente. Il espérait qu'en l'an 2000 une telle machine existerait.

([1]) Voir dans ce numéro *L'histoire des codes secrets.*

Math-Jeunes n° 110S, 3–6, 2005

Le Master Mind

Nadège Vandenabeele

Comme chaque dimanche après-midi, Géo enfourche son vélo et retrouve son ami Algo qui est informaticien.

Algo : J'ai acheté un cadenas à code pour ton magnifique vélo. Peux-tu en découvrir la combinaison ?

Géo : Mais il y a 4 chiffres. Cela veut dire qu'il y a 10 000 possibilités. Cela va me prendre beaucoup de temps !

Algo : Je te donnerai des indices à chaque essai, comme au *Master Mind*. Connais-tu ce jeu ?

Géo : C'est un jeu de société où un joueur doit deviner la combinaison secrète d'un autre joueur.

Algo : Exactement. J'ai envie d'informatiser ce jeu. Tu réfléchis avec moi ?

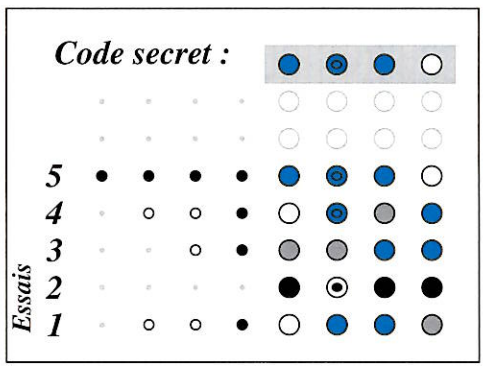
Géo : Dès que l'on parle d'informatique, je suis partant. Mais je ne me rappelle plus très bien les règles du jeu.

Algo : Les voici :

- *À l'origine, les joueurs ont à leur disposition des pions de six couleurs différentes. Ici, les « couleurs » sont les suivantes :*

blanc *gris* *bleu* *noir* *blanc-cerclé* *bleu-cerclé*

- *Le premier joueur, nous l'appellerons le « codeur », aligne quatre pions que le second joueur ne voit pas (bien qu'ils soient dévoilés sur la figure ci-dessous).*
- *Le deuxième joueur, nous dirons « le décodeur », a droit à 12 tentatives pour deviner le code secret.*
- *Après chaque essai, le codeur indique le nombre de pions bien placés (sans donner leur position) et le nombre de pions de la bonne couleur mais mal placés à l'aide du codage suivant : en regard de la proposition du décodeur, il plante dans la partie gauche du jeu un petit pion noir pour chaque pion de la bonne couleur et bien placé et un petit pion blanc pour chaque pion de la bonne couleur mais mal placé.*



Géo : En fait, si on remplace les couleurs par des chiffres, tu pourrais me faire deviner le code du cadenas !

Algo : Exactement, outre le fait que sur ton cadenas il y a 10 possibilités ! Dans la suite, les 6 couleurs seront remplacées par les chiffres de 1 à 6 et il n'y a que 6^4 possibilités.

Voici, une simulation du jeu. Je suppose que le code à découvrir est le 1142.

Essai n°	Essais	Bien Placés	Mal Placés
1	1234	1	2
2	1224	1	2
3	1242	3	0
4	1142 Gagné !	4	0

Géo : Le jeu en lui-même n'est pas difficile. Quel serait le rôle de l'ordinateur ?

Algo : Il pourrait tenir le rôle du codeur, en choisissant aléatoirement le code à découvrir, puis en analysant successivement chaque proposition du décodeur afin de lui indiquer le nombre de pions biens placés et le nombre de pions de la bonne couleur, mais mal placés.

Géo : C'est surtout l'analyse des combinaisons proposées qui ne me semble pas si évidente à réaliser !

Algo : Ne sois pas défaitiste. Nous allons décanter cela progressivement. Commençons par écrire la combinaison à deviner dans un tableau que nous appellerons `CODE`.

`CODE`= | 1 | 1 | 4 | 2 |

Ainsi, `CODE`(1) désigne le premier chiffre du tableau : `CODE`(1) = 1. `CODE`(2) désigne le deuxième chiffre du tableau : `CODE`(2) = 1, etc.

`TEST`= | 1 | 2 | 3 | 4 |

Géo : Ah oui ! Et la combinaison proposée par le décodeur sera stockée dans un autre tableau appelé `TEST`. Par exemple, `TEST`= | 1 | 2 | 3 | 4 |.

Algo : Tout à fait ! Maintenant, il faut analyser cette combinaison. Cela va se faire en plusieurs étapes :

1. *Dire combien de chiffres sont bien placés.*
2. *Connaître le nombre d'occurrences de chaque chiffre dans les deux tableaux.*
3. *Déterminer le nombre total de chiffres trouvés : ce sera le minimum des occurrences de chaque chiffre dans les deux tableaux.*
4. *Calculer le nombre de chiffres mal placés en remarquant qu'il est égal à la différence entre le nombre de chiffres trouvés et le nombre de chiffres bien placés.*

Une « occurrence » d'un élément dans un tableau est une position du tableau où cet élément figure. Le nombre d'occurrences d'un chiffre dans un des tableaux est donc le nombre de présences du chiffre dans le tableau.

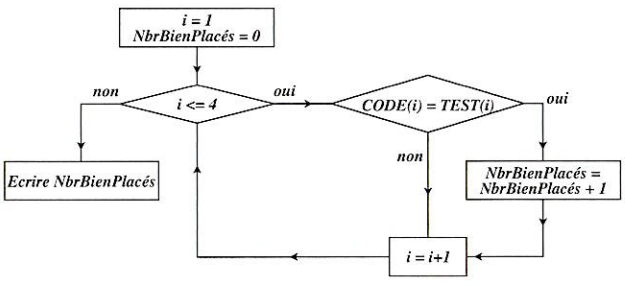
Mais, commençons par le commencement, et voyons comment compter le nombre de chiffres bien placés.

Géo : Je crois que j'ai une idée. il suffit de comparer les chiffres qui se trouvent à la même position dans les deux tableaux.

Algo : Autrement dit, il faut vérifier si `CODE`(1) = `TEST`(1), `CODE`(2) = `TEST`(2), etc. A chaque fois que c'est vrai, on augmente un compteur `NbrBienPlacés` de 1. Nous pouvons écrire tout cela de la façon suivante :

Au lieu de « augmenter un compteur de 1 », on dit souvent « *incrémenter* un compteur ».

```
Pour i de 1 à 4,
Si CODE(i) = TEST(i)
Alors NbrBienPlacés
= NbrBienPlacés + 1
```



On utilise des losanges pour symboliser des conditions à vérifier (Si ...) et des rectangles pour les opérations à effectuer.

La boucle « Pour `i` de 1 à 4 » nécessite un second test : `i` est-il inférieur ou égal à 4 ? Si oui, on incrémente le compteur `i` et on recommence, sinon on écrit le résultat.

Géo : Pas mal ton organigramme... il suffit de suivre les flèches ! On peut appliquer cela à l'exemple. Voici le tableau des résultats progressifs, juste avant d'incrémenter `i` de 1.

CODE	1	1	4	2
TEST	1	2	3	4

i	NbrBienPlacés
1	1
2	1
3	1
4	1

Algo : Tu as tout compris. Je t'en félicite ! On peut continuer avec la deuxième étape de la résolution : déterminer le nombre d'occurrences de chaque chiffre dans les deux tableaux.

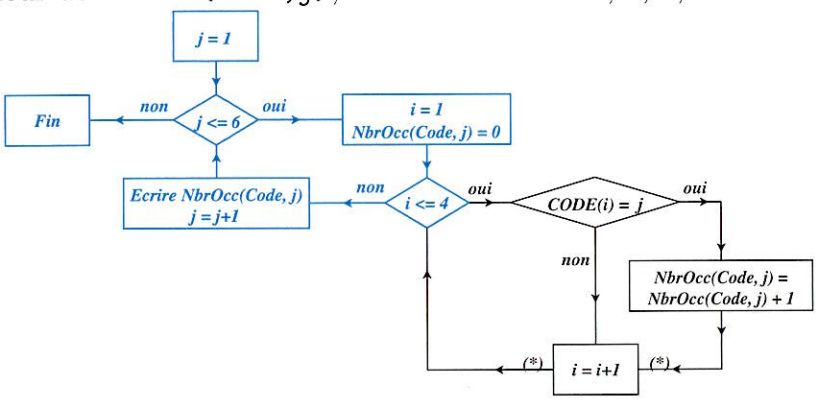
Géo : On veut obtenir quelque chose comme `NbrOcc(CODE,1)` qui désigne le nombre d'occurrences du chiffre 1 dans `CODE` (ici 2). Et il faut faire cela pour les six chiffres (de 1 à 6) et pour les deux tableaux. On aura donc une boucle contrôlant un indice `j` variant de 1 à 6. Elle apparaît en bleu sur l'organigramme ci-dessous.

j	i	NbrOcc(CODE,j)
1	1	1
	2	2
	3	2
	4	2
	5	2
2	1	0
	2	0
	3	0
	4	1
	5	1
3	etc	

Algo : Pour chaque valeur de `j`, il faut alors comparer chaque chiffre aux quatre chiffres de `CODE` puis de `TEST`.

Géo : Pour déterminer `NbrOcc(CODE,1)` on aura encore une boucle qui fera varier l'indice `i` du tableau de 1 à 4 et qui incrémentera le nombre d'occurrences de 1 dans le tableau `CODE`.

Algo : Oui, ensuite, il faut aussi faire varier les chiffres que l'on compare. On aura donc une deuxième boucle, pour `j` de 1 à 6. C'est exactement le même genre de boucle que précédemment ! Le tableau ci-contre montre le fonctionnement de cet organigramme au moment où l'on passe par (*). Chaque fois que le compteur `i` prend la valeur 5, l'ordinateur écrit la valeur de `NbrOcc(CODE,j)`, soit dans notre cas, 2, 1, etc.



Cet algorithme nécessite 24 (6×4) comparaisons du type « `CODE(i)=j ?` ». Si le nombre de chiffres du code et le nombre de chiffres à tester étaient très grands, il y aurait intérêt à adopter une procédure différente : le chiffre occupant la position `i` du tableau `CODE` étant `CODE(i)`, pour chaque valeur de `i` on augmente de 1 le compteur `NbrOcc(CODE,CODE(i))`. L'essentiel de l'algorithme est alors constitué d'une seule boucle, constituée d'une seule instruction `NbrOcc(CODE,CODE(i))= NbrOcc(CODE,CODE(i))+1`.

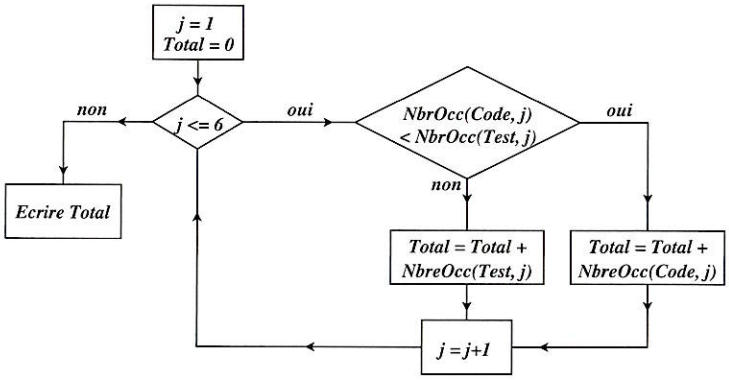
Géo : Ouaouh ! C'est performant tout ça ! Et dire qu'il faut faire la même chose pour le tableau `TEST`.

Algo : Il suffit de faire un « copier-coller » et le tour est joué. Le tout c'est de bien comprendre.

Géo : Je crois que je t'ai suivi, on peut continuer.

Algo : Puisqu'on connaît le nombre d'occurrences de chaque chiffre dans chaque tableau, on peut calculer le nombre de chiffres trouvés.

Par exemple, le nombre de « 1 » trouvés est le minimum de `NbrOcc(CODE,1)` et `NbrOcc(TEST,1)`. On fait cela pour tous les chiffres (`j` va donc de 1 à 6) et en additionnant chacun des minima, on obtient le nombre total de chiffres trouvés. C'est ce que fait l'organigramme ci-contre.



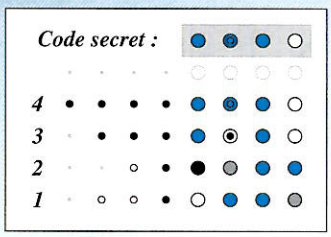
Géo : Et maintenant, on a presque fini. Il suffit de remarquer que le nombre de chiffres mal placés est la différence entre le total des chiffres trouvés et le nombre de chiffres bien placés.
Algo : Exactement, le décodeur dispose d'indices qu'il peut exploiter pour proposer une nouvelle combinaison. L'ordinateur analysera celle-ci et indiquera à nouveau le nombre de chiffres bien et mal placés. Le jeu continue jusqu'à ce que le second joueur trouve le code... ou abandonne !
Géo : C'est chouette mais pas facile du tout.
Algo : Je n'ai jamais dit que l'informatique était une science facile. Tout est basé sur l'analyse. Si tu as un bon esprit d'analyse, un bon esprit mathématique, alors tu as l'esprit informatique !
Géo : Et dire que lorsque nous jouons sans ordinateur, c'est notre cerveau qui fait tout ce travail sans même que nous nous en rendions compte !
Algo : A ton avis, comment le distributeur automatique de la banque vérifie-t-il ton code secret ? Sachant que lorsque ce code est erroné, il te le redemande. Mais tu n'as droit qu'à trois essais avant que ta carte soit avalée.
Géo : Très bonne question ! C'est sensiblement le même principe que celui du Master Mind ! Mais je dois rentrer maintenant. Puis-je te faire part de ma solution prochainement ?
Algo : Oui, bien sûr, réfléchis, prends ton temps !
Géo : Pas de problème ! Je serai au rendez-vous ! Encore merci pour le cadenas ! A bientôt.

Ami lecteur, si, comme Géo tu peux proposer une solution au problème posé par Algo, envoye-la par courrier postal à la SBPMef, 15 rue de la Halle à 7000 Mons, ou par e-mail à l'adresse `sbpm@sbpm.be`*, avant le 15 février 2005. N'oublie pas de mentionner tes coordonnées (nom, prénom, adresse, école, classe). A bientôt, dans le prochain Math-Jeunes pour la réponse !*

Une question de stratégie

Dans la partie décrite à la page 3, le décodeur ne tient pas compte de toutes les informations dont il dispose déjà. Par chance, il trouve la solution en cinq essais.
Bien sûr, le décodeur ne peut effectuer son premier essai que « au hasard ». À partir du deuxième essai, une stratégie souvent efficace consiste à tester une hypothèse.
Vu le résultat du premier essai, on peut supposer par exemple que le code secret comprend deux pions bleus, dont un bien placé, et un pion gris mal placé.

On peut alors compléter par un pion d'une couleur ne figurant pas dans la première ligne. Conservant l'hypothèse « deux pions bleus dont un bien placé », la partie pourrait se réaliser conformément à la figure ci-contre. Cette fois chaque proposition est compatible avec les informations déjà connues. De plus si l'hypothèse est vérifiée, il n'y a qu'une possibilité pour la quatrième ligne.



Math-Jeunes n° 110S, 7–8, 2005

Des codages au quotidien

Denis Pierson

Lorsqu'on nous communique un nombre, comment vérifier qu'il n'y a pas d'erreur ? À la transmission d'un nombre, les erreurs possibles sont

- transmettre un chiffre erroné,
- transmettre un chiffre illisible,
- intervertir deux chiffres (la plupart du temps consécutifs).

Il s'agit de détecter une erreur possible ; les méthodes de vérification devront donc essentiellement tenir compte de ces trois types d'erreur. Elles sont généralement basées sur le calcul d'un reste d'une division par un nombre N astucieusement choisi. Ce reste, appelé parfois « clé » ou « contrôle », voire « checksum » est adjoint au nombre N.

Dans cet article nous examinerons les codes utilisés pour les numéros de comptes bancaires.

Les numéros de comptes bancaires

Un numéro de compte bancaire, par exemple 001-0314779-90, comporte trois parties. La première (3 chiffres), 001, est le numéro de la banque. La seconde (7 chiffres), 0314779, est le numéro de compte proprement dit. La troisième (deux chiffres), 90, sert de contrôle. Il est le reste de la division du nombre 0 010 314 779, constitué des deux premières parties, par un nombre N. Ce nombre N doit être inférieur à 100 puisque le reste de la division ne peut avoir que deux chiffres.

Pour déterminer si un numéro de compte est valide, on effectue cette division et on compare le reste à la troisième partie. S'il y a égalité, le numéro est valide. Mais cela ne veut pas dire qu'il est correct : lorsqu'on transmet un numéro à un correspondant, il peut se produire une ou plusieurs erreurs aboutissant au remplacement d'un numéro valide par un autre également valide. Le problème est de choisir N de façon à éviter le plus possible une telle situation.

S'il y a erreur sur un seul chiffre, notons d la différence entre le chiffre correct et le chiffre erroné. La différence entre les deux codes, correct et erroné, est alors du type $10^n.d$, avec $0 \leqslant n \leqslant 9$. Si le nombre N divise $10^n.d$, les divisions par N des deux codes ont le même reste et l'erreur n'est pas détectée. On doit donc éviter de choisir pour N un diviseur du produit $10^n.d$. Comme un nombre premier ne divise un produit que s'il divise un des deux facteurs, on va choisir pour N un nombre premier qui ne divise ni 10^n, ni aucun nombre d'un seul chiffre.

Si on a interverti deux chiffres a (de rang n) et b (de rang $m < n$), la différence entre le nombre exact et le nombre erroné vaudra $a.10^n + b.10^m - b.10^n - a.10^m$, soit $10^m.(a-b).(10^{n-m}-1)$. Il faudra aussi éviter de choisir pour N un diviseur de 99, 999...

Par ailleurs, plus le nombre N est grand, plus il y a de clés de contrôle différentes et donc plus grande est la probabilité de détecter une erreur. En pratique le plus grand nombre premier de deux chiffres, soit $N = 97$ satisfait aux diverses conditions trouvées. Le nombre formé par les deux derniers chiffres est donc le reste de la division par 97, avec toutefois une exception ! Si le reste est nul (on a encore peur du zéro, voyez les ascenseurs...) on le remplace par 97.

Comme vérification, prenons le compte 001-0314779-90. Le quotient entier de 0 010 314 779 par 97 vaut 106 337 et le reste vaut $0\,010\,314\,779 - 106\,337 \times 97 = 0\,010\,314\,779 - 10\,314\,689 = 90$. Le numéro est correct.

(1) Denis Pierson était élève de rhétorique à l'Athénée Jean Rey de Couvin en 2003–2004.

On a récemment introduit l'IBAN (*International Bank Account Number*). On espérait enfin voir une vraie uniformisation des comptes bancaires avec un système de contrôle simple. Malheureusement, ce n'est pas le cas et chaque pays a adopté sa propre structure. Comment est construit cet IBAN ?

L'IBAN de la Société Belge des Professeurs de Mathématique d'expression française est
BE26 0000 7280 1429.

Il se compose d'abord de quatre caractères : deux lettres (la codification ISO (*International Organisation for Standardisation*) du pays, BE : Belgique, FR : France, CH : Suisse...) suivies de 2 chiffres qui constituent la clé de contrôle.

On poursuit avec l'identification de l'établissement financier (caractères alpha-numériques) suivi du BAN (*Bank Account Number*). Pour fixer les idées, supposons que mon numéro de compte soit : 001-0314779-90. Je suis belge (code BE) ; mon établissement financier est codé 001. Comment, à partir de ces données, reconstituer l'IBAN ? On écrit BE00 (le code du pays suivi provisoirement de 2 zéros à la place de la clé de contrôle) puis le code de l'établissement financier suivi du numéro de compte, ce qui donne : BE00001031477990.

En Belgique les codes des établissements financiers sont exclusivement numériques (3 chiffres). Ainsi, « 000 » désigne un compte ouvert à la Banque de la Poste, « 001 » désigne un compte Fortis, etc. Les fusions de banques ont eu pour conséquence que certaines d'entre elles disposent de plusieurs codes différents.

Il reste à calculer les deux chiffres de contrôle figurant après le code du pays. Pour cela on transpose les 4 premiers caractères vers la fin, ce qui donne : 001031477990BE00 Malheureusement des lettres figurent dans le code du pays et parfois dans le numéro de compte bancaire. On les remplace avec la règle A→10, B→11, C→12... Z→35. Cela nous donne : 001 031 477 990 111 400 Ensuite on effectue, comme anciennement une division par 97. Le contrôle sera égal à 98 moins le reste de cette division entière.

On remarque que l'IBAN d'un numéro de compte bancaire belge comporte DEUX clés de contrôle : une belge et une internationale.

Dans l'exemple le quotient entier vaut 10 633 793 712 488 et le reste vaut 64. En effet $10\,633\,793\,712\,488 \times 97 = 1\,031\,477\,990\,111\,336$. Le contrôle est la différence entre 98 et ce reste, c'est-à-dire 34. Mon IBAN serait donc : BE34001031477990. Pour plus de facilité, cet IBAN est décomposé en tranches de 4 caractères, c'est-à-dire : BE34 0010 3147 7990.

Les Codes à barres

Il existe différents types de codes à barres. Tout le monde connaît ceux qui sont apposés sur les articles vendus dans le commerce. Les codes ISBN (*International Standard Book Number*) servent à identifier tous les livres édités dans le monde entier. Ils sont également accompagnés de codes à barres. Le cas de ces codes a été traité précédemment dans un article de J. Miewis, *Le code à barres EAN* (*Math-Jeunes* n°32 (1986)). Cet article peut être téléchargé sur le site de la SBPMef (adresse : `http://www.sbpm.be`). Il a de plus été reproduit sur le CD-Rom diffusé avec le n° 106 de notre revue. Des exemplaires de ce CD-Rom peuvent encore être obtenus en s'adressant au secrétariat de la SBPMef. (NDLR)

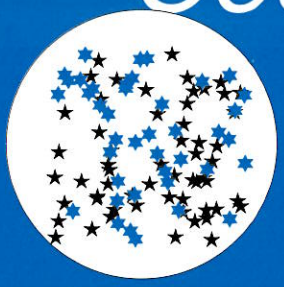
Sources :

Encyclopaedia Universalis.
Encyclopédie Microsoft Encarta.
http ://users.pandora.be/worldstandards/barcodes.htm
www.encyclobd.com/
www.galaxia.be/NOR010.htm
www.gomaro.ch/isbnissn.htm
homepages.cwi.nl/ dik/english/codes/isbn.html
mitglied.lycos.de/buran/knowhow/codes/ISBN.html
flagspot.net/flags/bib-lau.html

Math-Jeunes n° 110S, 9–10, 2005

Codages de nombres

Yolande Noël-Roch



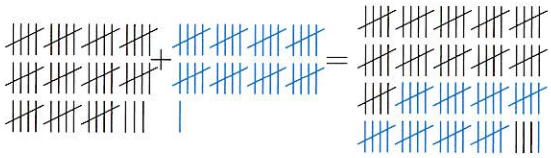
Dans l'usage courant, l'écriture 402 523 évoque le nombre quatre cent deux mille cinq cent vingt-trois. Cela reflète notre habitude de fonctionner dans une numération de position de base dix. Deux propriétés fondamentales interviennent :

- **la place d'un chiffre détermine sa valeur**
- **l'usage du zéro**

Dans la base dix, les chiffres sont 0, 1, 2, ... 9 et, selon leur rang de droite à gauche, ils désignent un nombre d'unités, de dizaines, de centaines, ...
Ainsi, en base dix :
523 = 3 + 2 dizaines + 5 centaines.
Mais la même écriture donnée par un habitant d'une planète où l'on utilise la numération octale (de base huit) a une autre signification :
523 = 3 + 2 huitaines + 5 soixante-quatre-aines.

1. À chacun sa façon

Des confetti sont répandus sur la table et quatre amis font un dénombrement dont voici le résultat :



	noir	bleu	total
Barnabé	Voir ci-contre		
Thierry	2011	1112	10200
Didier	58	41	99
Céline	3A	29	63

soit, pour les Romains, IC.

Chacun des quatre amis a dénombré les confetti noirs et les confetti bleus et a ensuite additionné les deux nombres obtenus ... les quatre totaux sont donc les mêmes, malgré les apparences ! Comment s'en convaincre ?

1.1 Barnabé et Didier

Barnabé estime qu'il est plus facile de compter des objets rangés que des objets éparpillés. Il les range donc et, pour faciliter ensuite le comptage, il les groupe par cinq parce que la litanie « cinq, dix, quinze, ... » lui est familière.
Didier a compté sans réfléchir « un, deux, ... » jusqu'à cinquante-huit confetti noirs, puis quarante et un confetti bleus et enfin ajouté sans problème les deux nombres obtenus.

Extrait de *Le théorème de Morcom* par Goffin et Peeters, Ed. Les humanoïdes associés

Nous pouvons donner une même structure aux deux démarches :

- Barnabé : $(11 \times 5 + 3) + (8 \times 5 + 1) = 19 \times 5 + 4$.
- Didier : $(5 \times 10 + 8) + (4 \times 10 + 1) = 9 \times 10 + 9$.

1.2 ... et les Autres

Comme Mr. Jourdain écrivait en prose sans le savoir, Didier a travaillé **en base dix** ... comme il le fait toujours ! Qu'ont bien pu inventer Thierry et Céline pour se distinguer ?

Thierry a décidé de travailler **en base trois**. Au lieu d'utiliser *unité, dizaine, centaine, ...*, il utilise *unité, troizaine, neuvaine, ...* et le total 10200 qu'il écrit désigne, dans notre coutume décimale

$$0 + (0 \times 3) + (2 \times 9) + (0 \times 27) + (1 \times 81)$$

ou encore, en marquant mieux l'utilisation de la base 3 :

$$0.3^0 + 0.3^1 + 2.3^2 + 0.3^3 + 1.3^4.$$

Les calculs informatisés sont exécutés en base 2 (on parle de « calcul binaire »). En effet, les signes **0** et **1** (les « bits ») suffisent pour distinguer si un circuit est ouvert ou fermé. En binaire, la suite des naturels est **0**, **1**, **10**, **11**, **100**...

Voici les tables d'addition et de multiplication :

+	**0**	**1**
0	**0**	**1**
1	**1**	**10**

×	**0**	**1**
0	**0**	**0**
1	**0**	**1**

Mais ce codage nécessite de longues chaînes de bits **0** et **1** pour écrire les nombres, rendant malaisées leur lecture et l'appréciation de leur ordre de grandeur.

C'est pourquoi les résultats sont fournis généralement

- soit en base 8. Chaque chiffre octal correspond à un groupe de trois bits :

000	**001**	**010**	**011**
0	1	2	3
100	**101**	**110**	**111**
4	5	6	7

- soit en base 16. Chaque chiffre hexadécimal correspond à un groupe de quatre bits :

0000	**0001**	**0010**	**0011**
0	1	2	3
0100	**0101**	**0110**	**0111**
4	5	6	7
1000	**1001**	**1010**	**1011**
8	9	A	B
1100	**1101**	**1110**	**1111**
C	D	E	F

Exemple : Traduisons le binaire **1011101111**

- en octal :

1 **011** **101** **111**
1 3 5 7

- en hexadécimal :

10 **1110** **1111**
2 *E* *F*

En tout, Thierry a bien dénombré nonante-neuf confetti. Vous pouvez contrôler qu'il a aussi dénombré cinquante-huit confetti noirs et quarante et un confetti bleus.

Quant à Céline, elle défend l'usage d'une autre base mais vous savez qu'elle a compté (sans se tromper !) les mêmes nombres. Vous constatez qu'elle utilise au moins le chiffre A en plus des chiffres usuels et que, dans son total, $A + 9 = 13$. Vous pouvez découvrir la base qu'elle utilise !

2. Pour et/ou contre

Didier estime que Barnabé use beaucoup de papier et de crayon tout en restant clair mais que Thierry et Céline risquent fort d'être mal compris :

- la somme obtenue par Thierry risque de faire croire qu'il s'est trompé en additionnant ses deux résultats,
- rien de ce genre chez Céline mais qui interprétera correctement les dénombrements effectués ?

Nos deux amis ne se laissent pas impressionner et défendent leur point de vue :

- Thierry : J'écris $2011_3 + 1112_3 = 10200_3$ pour signaler que je travaille en base trois. Pour vous montrer pourquoi j'aime bien cette base, voici les tables d'addition et de multiplication ainsi que le calcul de la somme et du produit des deux nombres.

+	0	1	2
0	0	1	2
1	1	2	10
2	2	10	11

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	11

```
  2 0 1 1
+ 1 1 1 2
---------
1 0 2 0 0
```

```
        2 0 1 1
×       1 1 1 2
---------------
      1 1 0 2 2
    2 0 1 1
  2 0 1 1
2 0 1 1
---------------
1 0 0 2 1 0 0 2
```

- Céline : Tes tables d'addition et de multiplication me séduisent mais, pour ma part, j'ai un copion des tables d'addition et de multiplication en base seize. (Voir la rubrique « Jeux ».)

  Tu écris tous tes nombres à l'aide des trois chiffres 0, 1 et 2. Pour ma part, je calcule en hexadécimal (base seize) : j'utilise les seize chiffres 0, 1, 2, 3, ⋯, 9, A, B, C, D, E, F. Voici ma version des nombres :

  $$2011_3 = 3A_{16} \text{ et } 1112_3 = 29_{16}$$

  et le calcul de leur somme et de leur produit.

```
  3 A
+ 2 9
-----
  6 3
```

```
  3 A
× 2 9
-----
2 0 A
7 4
-----
9 4 A
```

D'une manière générale, le codage en base b utilise b chiffres différents (en écriture hexadécimale, ce sont 0, 1, 2, ... 9, A, B, C, D, E et F) et

$$(a_n \cdots a_1 a_0)_b = a_0 + (a_1 \times b) + \cdots + (a_n \times b^n).$$

Math-Jeunes n°110S, 11–14, 2005

Le contrôle des erreurs dans les informations digitales

Pierre Dufour

Introduction

Dans la société de l'information qui est la nôtre aujourd'hui, l'information est de plus en plus souvent représentée sous forme digitale. A titre d'exemples :

- Notre bon vieux téléphone (*Plain Old Telephone System*) ne transmet plus la voix sous forme analogique. Il la transforme en une séquence de 0 et de 1 et utilise des centraux digitaux plutôt que des centraux analogiques.
- Les protocoles de communication utilisés par Internet sont de type digital.
- Les appareils photographiques digitaux enregistrent les photos comme une série de points. Ils remplacent de plus en plus souvent les appareils classiques.
- Dans les imprimantes, les caractères gravés ont été depuis longtemps remplacés par des matrices de points.
- À la différence des disques noirs (en vinyle), les CD-ROMs et les DVD enregistrent les données sous forme digitale.

Vu la quantité d'informations manipulées et la vitesse de traitement, il est essentiel que les systèmes de stockage de données et les systèmes de transmission de données se chargent d'un contrôle d'erreurs. Contrairement à la représentation sous forme analogique qui était la règle dans le passé, la représentation sous forme d'une séquence de **0** et de **1** permet un contrôle facile de la vraisemblance d'une configuration de bits.

Dans certains cas, on se contente d'alerter l'utilisateur lorsqu'une erreur se produit. On utilise dans ce cas un code détecteur d'erreurs qui se borne à constater qu'une erreur s'est produite et à avertir que les données ne sont pas fiables. C'est le cas lors de la lecture sur disque ou lors de la transmission de données sur des lignes téléphoniques.

Dans d'autres cas (par exemple lors des transferts de données entre la mémoire de l'ordinateur et les registres du processeur), on ne se contente pas de détecter l'erreur mais on souhaite qu'elle soit automatiquement corrigée.

Les systèmes que nous présentons ci-dessous sont adaptés aux situations dans lesquelles il y a peu d'erreurs. En fait, on suppose qu'au plus une erreur s'est produite dans la séquence testée. S'il y a risque de plusieurs erreurs, des systèmes plus élaborés doivent être utilisés. Sur les CD-ROMS, on combine des codes détecteurs d'erreurs et des codes correcteurs d'erreurs pour permettre un meilleur contrôle.

1. La détection d'erreurs

Un émetteur et un destinataire peuvent construire un code détecteur d'erreurs en convenant de faire suivre l'information transmise (une séquence de bits) par un bit supplémentaire, appelé « bit de contrôle » et de valeur telle que le nombre total de bits dans l'état **1** soit toujours **impair**. Si, lors d'une transmission, un seul bit est altéré, le destinataire peut s'en rendre compte puisque le nombre de bits dans l'état **1** est devenu pair (par suite de la transformation d'un **0** en un **1** ou d'un **1** en un **0**). Il détecte ainsi qu'une erreur de transmission a eu lieu et peut mettre le message en quarantaine ou demander sa retransmission.

Par exemple, la séquence **11001010** sera complétée par un **1**, donnant ainsi la séquence **110010101**. L'altération d'un seul bit lors de la transmission pourra être détectée puisque le nombre de bits dans l'état **1** dans le message complété reçu sera devenu pair... Cette méthode est utilisée pour vérifier les données enregistrées sur disque et est à la base des « parity errors » qu'il vous est peut-être déjà arrivé de rencontrer.

Lors de la transmission de données sur une ligne téléphonique, on utilise un code CRC (Cyclic Redundancy Check) (voir [3]) qui est basé sur l'insertion de plusieurs bits de contrôle et, dans de nombreux cas, permet de détecter plusieurs erreurs.

2. La correction d'erreurs

Un système plus élaboré(1) permet non seulement de détecter une erreur (unique) mais également de déterminer la position du bit erroné, ce qui permet de le corriger.

Soit un octet (8 bits) à transmettre, **11001010** par exemple. Nous le plaçons, tel quel, à la première ligne du tableau ci-dessous. Nous numérotons les bits de gauche à droite.

Octet	1	1	0	0	1	0	1	0
Numéro	1	2	3	4	5	6	7	8

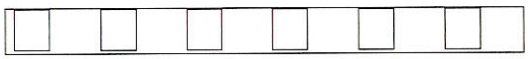
En binaire(2), les nombres de 1 à 8 s'écrivent **1**, **10**, **11**, **100**, **101**, **110**, **111**, **1000**. Du fait de la présence du nombre 8, nous avons besoin de quatre bits de contrôle (et nous utilisons les écritures en quatre chiffres : **0001**, **0010**, **0011**, **0100**, **0101**, **0110**, **0111**, **1000**). Ces quatre bits doivent être transmis en même temps que l'octet. C'est donc un message de douze bits qui doit être transmis, ce qui nous amène à ajouter quatre colonnes au tableau et à renuméroter les bits de l'octet. Nous réservons les numéros 1, 2, 4 et 8 aux bits de contrôle.

Message	?	?	1	?	1	0	0	?	1	0	1	0
Numéro	1	2	3	4	5	6	7	8	9	10	11	12

L'octet à transmettre occupe les positions n° 3, 5, 6, 7, 9, 10, 11 et 12 du tableau. Il reste à déterminer les bits de contrôle.
Le bit n° 1 du message va servir à déterminer le bit des unités du numéro d'un éventuel bit erroné. On remarque que les nombres compris entre 1 et 12 dont le bit des unités vaut **1** sont 1, 3, 5, 7, 9 et 11. On donne la valeur **1** ou **0** au bit n° 1 du message de manière telle que parmi les positions 1, 3, 5, 7, 9 et 11 figurent un nombre impair de bits égaux à **1**.
Dans notre exemple, on trouve des **1** aux positions 3, 5, 9 et 11. On donnera donc la valeur **1** au bit n° 1.
À la réception du message, on vérifie si les positions de numéros 1, 3, 5, 7, 9, 11 comportent un nombre impair de **1**. Si c'est le cas, le bit des unités du numéro du bit erroné vaut **0**, sinon il vaut **1**.
On choisit de façon analogue les autres bits de contrôle. Pour simplifier le travail, complétons le tableau précédent en écrivant verticalement dans chaque colonne le numéro de cette colonne traduit en binaire (quatre chiffres).

Message	?	?	1	?	1	0	0	?	1	0	1	0
Numéro	1	2	3	4	5	6	7	8	9	10	11	12
Huitaines	0	0	0	0	0	0	0	1	1	1	1	1
Quatraines	0	0	0	1	1	1	1	0	0	0	0	1
Deuzaines	0	1	1	0	0	1	1	0	0	1	1	0
Unités	1	0	1	0	1	0	1	0	1	0	1	0

Les quatre dernières lignes du tableau précédent servent à construire des « masques » comportant des « fenêtres » à l'emplacement des bits égaux à **1**. Par exemple à la ligne des unités correspond le masque



Dans chaque ligne du tableau ci-dessous, un des quatre masques est placé sur le message à transmettre. Le bit ? d'une ligne est déterminé de façon qu'un nombre impair de **1** apparaissent dans les fenêtres de la ligne.

Huitaines	?	?	1	?	1	0	0	?	1	0	1	0
Quatraines	?	?	1	?	1	0	0	?	1	0	1	0
Deuzaines	?	?	1	?	1	0	0	?	1	0	1	0
Unités	?	?	1	?	1	0	0	?	1	0	1	0

En conséquence, le message transmis est

1	1	1	0	1	0	0	1	1	0	1	0

(1) Cette présentation est reprise d'un article paru dans *Math-Jeunes*, [1].
(2) Voir dans ce numéro l'article « Codages de nombres ».

Notons en passant que les bits n° 1 et 2 du message contrôlent 4 bits de l'octet à transmettre tandis que les bits n° 4 et 8 en contrôlent 5.
Voyons maintenant ce qui se passe si un bit est mal transmis, par exemple le bit n° 10 : lors du contrôle à la réception, les masques relatifs aux huitaines et aux deuzaines détecteront une erreur sur les bits qu'ils contrôlent... Le numéro du bit erroné est donc (en binaire) **1010**, soit 10. D'une manière générale, si une seule erreur de transmission a lieu, le n° du bit erroné est donné par la somme des numéros des bits de contrôle pour lesquels on détecte une erreur de parité. Notons que l'erreur de transmission pourrait porter sur un des bits de contrôle eux-mêmes. Dans ce cas, un seul masque détectera l'erreur.
En conclusion, on dispose d'un moyen de détecter le bit en erreur et donc de le corriger...

3. Combien de bits de contrôle ?

Demandons-nous maintenant s'il est possible de trouver une formule générale permettant de déterminer le nombre r de bits de contrôle qu'il faut accoler à un message de longueur m pour corriger une erreur de transmission.

Dans l'exemple ci-dessus, nous contrôlions 8 bits avec 4 bits de contrôle. Il est facile de se convaincre que les 4 bits de contrôle nous auraient permis de contrôler 11 bits. En effet, avec 4 bits, il est possible de représenter un nombre allant jusqu'à 15 (8+4+2+1) et il reste 11 bits (15-4 bits de contrôle) pour le message original. Avec 5 bits de contrôle, nous pourrons de la même manière contrôler un message original comportant 26 bits. D'une manière générale, avec r bits on peut représenter un nombre allant jusqu'à $2^r - 1 = \underbrace{\mathbf{1\ldots1}}_{r \text{ bits}}$, donc le nombre r de bits nécessaires pour contrôler un message original comportant m bits est le plus petit r qui satisfait l'inéquation suivante :

$$m + r \leqslant 2^r - 1 \qquad (1)$$

Dans ce qui suit, nous montrerons qu'il n'est pas possible de construire un code correcteur d'erreurs comportant un nombre de bits de contrôle moindre que celui donné ci-dessus.

Pour cela, on aborde le problème d'une autre manière :

On utilise les mêmes notations que ci-dessus : m est le nombre de bits du message original à transmettre, r est le nombre de bits de contrôle. Le message complété comportera donc $m+r$ bits. Il est facile de se rendre compte que 2^m messages originaux différents pourront être transmis. Comme un message original ne peut être complété que d'une seule façon en un message complété valide, parmi les 2^{m+r} messages complétés, seulement 2^m sont des messages complétés valides.

A partir d'un message complété valide, les erreurs de transmission (supposées ne se produire que sur un bit) pourront engendrer $m+r$ messages altérés différents.

Pour fixer la terminologie, nous appellerons dorénavant :
- *message original* : un message qu'on souhaite transmettre et qui comporte m bits ;
- *message complété* : un message original auquel on a ajouté r bits et qui en comporte donc $m+r$;
- *message complété valide* : un message de $m+r$ bits obtenu en calculant les r bits de contrôle d'un message original et en les adjoignant à celui-ci ;
- *message complété altéré* : un message de $m+r$ bits différant par un bit d'un message complété valide ;
- *message complété non valide* : un message de $m+r$ bits qui n'est pas valide.

Si on veut pouvoir retrouver le message complété valide M_i qui a été transmis à partir d'un message complété altéré qui a été reçu, il faut que celui-ci ne puisse provenir que d'un seul message complété valide.

Deux ensembles sont dits *disjoints* s'ils n'ont aucun élément commun. On dit aussi que leur intersection est vide.

Chaque message complété valide M_i ainsi que les $m+r$ messages altérés qui lui sont associés (en tout, $m+r+1$ messages) constituent un sous-ensemble de l'ensemble des messages complétés qui doit donc être disjoint des sous-ensembles associés à tous les autres messages complétés valides $(M_j, j \neq i)$.

L'ensemble des 2^{m+r} messages complétés (valides, altérés et non valides) doit donc contenir au moins 2^m sous-ensembles disjoints deux à deux de $m+r+1$ messages complétés chacun et on a donc :

$$2^m(m+r+1) \leqslant 2^{m+r} \tag{2}$$

Après division par 2^m, on retrouve l'inéquation (1). Il est remarquable de constater que le code correcteur d'erreurs que nous avons présenté permet d'organiser les sous-ensembles de manière à ne pas perdre de place...

On peut reformuler ce qui a été écrit ci-dessus en disant que, si une seule erreur se produit lors de la transmission d'un message complété, le message complété altéré reçu différera du message complété valide qui a été transmis par un seul bit et il devra rester différent de tous les autres messages complétés valides par au moins deux bits (si ce n'était pas le cas, les sous-ensembles associés aux messages complétés valides ne seraient pas disjoints). On pourra donc retrouver le message original à partir d'un message complété dont un bit a été altéré.

On remarque que deux messages complétés valides diffèrent toujours l'un de l'autre par au moins trois bits. Le nombre de bits qui diffèrent entre deux messages de même longueur est appelé *distance de Hamming* et le code que nous avons présenté est appelé *code de Hamming*, (voir [2], [3]).

Pour en savoir plus

[1] **P. Dufour, J. Lion, Cl. Villers**, De l'autopsie d'un jeu à un code correcteur d'erreurs, *Math-Jeunes*, 48 (1990), 110–113.

[2] **R.W. Hamming**, Error Detecting and Error Correcting Codes, *Bell System Tech. Journal*, 29 (1950), 147–160.

[3] **A.S. Tanenbaum**, Computer Networks, *Prentice-Hall 2002.*

L'ordinateur quantique : la fin des cryptosystemes RSA ?

Une menace se profile à l'horizon pour la survie des cryptosystèmes RSA (voir dans ce numéro l'article de F. Valette) : c'est l'ordinateur *quantique* imaginé dès les années 70 par le prix Nobel de physique 1965 Richard Feynman (1918 – 1988). Longtemps considéré comme une vue de l'esprit, un tel ordinateur n'a d'abord guère excité les chercheurs. Son fonctionnement est basé sur les principes de la mécanique quantique et non sur ceux de l'électromagnétisme (comme nos ordinateurs actuels).

En 1994, Peter Shor, professeur au très prestigieux *Massachusetts Institute of Technology* (cet institut où le système RSA a été inventé en 1977), imagine pour la factorisation d'un naturel n un algorithme quantique ne nécessitant qu'un nombre d'opérations binaires polynômial au lieu d'exponentiel en $\ln(n)$, donc tout à fait performant en un temps raisonnable. Est-ce la fin des cryptosystèmes RSA ?

Actuellement, les spécialistes de la sécurité informatique peuvent être tranquilles : bien qu'un algorithme quantique existe déjà, on n'a pas encore réussi à... construire un ordinateur quantique pour le faire fonctionner ! Mais depuis 1994, quelques physiciens et informaticiens, séduits par la rapidité de calcul prévue, rêvent d'en fabriquer un. Jusqu'ici les performances obtenues ne sont pas extraordinaires : on a réussi à construire un ordinateur quantique qui a factorisé...15 en 3×5 ! Mention : « Peut faire mieux ! » Mais parfois les progrès techniques nous surprennent par leur rapidité. Alors qui sait ?

Math-Jeunes n° 110S, 15–18, 2005

De la scytale à Enigma

Julie Christophe et Julie De Saedeleer

Que ce soit pour des raisons militaires, pour des raisons commerciales ou pour d'autres raisons encore, il arrive qu'un message soit classé « confidentiel ». Il faut alors trouver une technique qui permette de le faire parvenir à son destinataire sans qu'il puisse être compris par un indiscret.

Une forme primitive d'assurer le secret d'un message est tout simplement de dissimuler le message lui-même. C'est ce qu'on appelle la *Stéganographie.*

Déjà dans des temps reculés, des procédés stéganographiques existaient : par exemple on écrit un message confidentiel à l'encre « sympathique » (souvent du jus de citron) entre les lignes d'un message anodin. Le message secret apparaît quand on échauffe le papier. On peut aussi raser la tête de quelqu'un et écrire le message sur le crâne. Quand les cheveux auront repoussé, le message sera dissimulé.

Des méthodes modernes reposent sur la modification de quelques pixels d'une image informatique. Mais quelle qu'elle soit, une méthode stéganographique n'est efficace que si l'« ennemi » ignore la présence d'un message !

Le principe de la *cryptographie* est différent : le message n'est pas dissimulé, mais il est rendu incompréhensible... L'émetteur et le destinataire doivent donc se mettent d'accord sur la façon de brouiller les futurs messages. Ils doivent inventer une **clé** et choisir la façon de s'en servir !

Comment brouiller un message ?

Deux méthodes simples se sont développées : la *transposition* et la *substitution.* Dans la transposition, les lettres composant le message sont simplement permutées entre elles : on remplace le message par une de ses anagrammes. Dans les méthodes par substitution, on remplace chaque lettre par un autre signe : soit une lettre du même alphabet, soit un symbole d'un autre jeu de caractères. Voyons quelques exemples à travers les âges.

1. La cryptographie dans l'antiquité.

Le cylindre

Au cinquième siècle avant J.C., les généraux spartiates (Grèce) étaient munis d'un baton cylindrique (appelé « *scytale* ») d'un diamètre déterminé. Ils prenaient un bandeau de parchemin long et étroit qu'ils enroulaient autour de leur bâton et sur lequel ils écrivaient le message. Le message était alors déroulé et envoyé au destinataire. La clé était donc le diamètre du bâton utilisé, qui assurait l'alignement correct des lettres. Cette méthode d'encodage est une méthode de transposition.

Le chiffre de César

La méthode utilisée par les armées de César est une substitution. C'est un décalage de n unités des lettres de l'alphabet avec $1 \leqslant n \leqslant 25$. Donnons comme exemple un décalage d'ordre 3 :

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
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

(1) En 2003–2004, les auteurs étaient étudiantes en mathématiques à l'U.L.B.

(2) Certaines illustrations sont exraites des ouvrages cités dans la bibliographie (voir en fin d'article).

Ce cher Jules a parfois employé un code plus simple encore, comme nous l'apprend ce passage de la *Guerre des Gaules* :
[César] apprend par des prisonniers ce qui se passe chez Cicéron et combien sa situation est périlleuse. Il décide alors un cavalier gaulois, par de grandes récompenses, à porter une lettre à Cicéron. Cette lettre qu'il envoie est écrite en caractères grecs, afin que l'ennemi, s'il l'intercepte, ne connaisse pas nos projets.
J. César, La Guerre des Gaules, Livre V, Chap. 49

Encodez le message « JE LIS MATH JEUNES » avec le code de César.

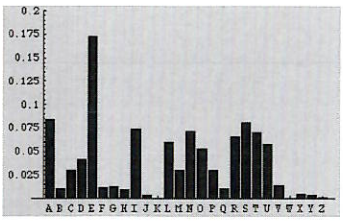
De façon plus générale, on peut également permuter les lettres aléatoirement, ce qui veut dire que le décalage n'est plus constant et la clé est une permutation des lettres de l'alphabet :

$$\begin{pmatrix} a & b & c & d & e & f & g & h & i & j & k & l & m & n & o & p & q & r & s & t & u & v & w & x & y & z \\ f & m & q & s & a & z & u & n & w & c & x & r & g & y & o & b & e & j & t & h & l & d & i & k & p & v \end{pmatrix}$$

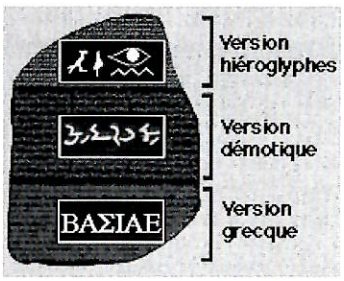
2. La cryptanalyse

Si les cryptographes codent les messages, en réponse, naissent les cryptanalystes. La cryptanalyse est la science des décodages des messages. Elle évolue en fonction des avancées de la cryptographie. Dès que la clé d'un code a été percée par les cryptanalistes, les cryptographes inventent une nouvelle manière de crypter fournissant un nouveau challenge aux cryptanalistes. C'est donc une guerre sans fin.

La méthode des fréquences



Cette méthode a permis de « casser » le code de César. Effectivement, chaque lettre de texte clair est toujours codée par la même lettre de texte chiffré. Chaque lettre de l'alphabet a une certaine fréquence selon la langue utilisée.([3]) Si le message est suffisamment long et que les lettres qui le composent respectent cette répartition, on peut le déchiffrer en comparant les deux tables de fréquences.



Casser un code est beaucoup plus facile quand on dispose d'un texte en deux versions : une version « en clair » et une autre encodée. Ainsi la découverte de la Pierre de Rosette a permis de déchiffrer les hiéroglyphes grâce à la comparaison entre trois versions du même texte en hiéroglyphes, en démotique et en grec.

Découvert en Crète en 1908, le disque de Phaistos qui date environ de 1800 à 1600 avant J.C. n'est toujours pas déchiffré. Le texte n'est pas assez long pour pouvoir appliquer la méthode des fréquences.

3. Les limites de la méthode des fréquences

Le chiffre de Vigenère

Ce chiffre a été créé par Blaise de Vigenère au XVI^{e} siècle. Pendant deux siècles, il résista aux attaques des cryptanalystes.

([3]) En français, les lettres les plus fréquentes sont dans l'ordre, E, S, I A, N, T, U, R...

L'efficacité de ce code repose sur l'utilisation de 26 alphabets dans un carré, dit de Vigenère. En haut se trouve l'alphabet ordinaire et en dessous s'alignent 26 alphabets, chacun décalé d'une lettre par rapport au précédent.

	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
1	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	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	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
26	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

Pour chiffrer et déchiffrer un message, Blaise de Vigenère invente un système de mot clé. Voyons un exemple : Nous allons crypter le texte « bonjour aux lecteurs de Mathjeunes » avec le mot clé CRYPTOGRAPHIE. Nous écrivons le texte et en dessous une succession du mot clé de même longueur que le texte (voir ci-dessous).

Texte clair : b o n j o u r a u x l e c t e u r s d e m a t h j e u n e s
Mot clé : C R Y P T O G R A P H I E C R Y P T O G R A P H I E C R Y P
Texte crypté : D F L Y H I X R U M S M G V V S G L R K D A I O R I W E C H

Pour crypter, on prend la lettre située à l'intersection de la colonne de la lettre en clair et de la ligne dont l'alphabet commence par la lettre du mot clé. Pour la première lettre, b, on regarde l'intersection entre la colonne b et la ligne commençant par C. La lettre chiffrée est donc D.

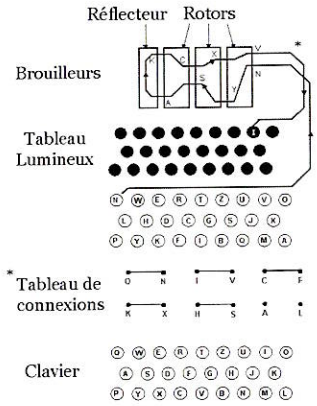
4. La machine Enigma

Histoire

Dans les années 1920, Arthur Scherbius, un Hollandais et Richard Ritter, un Allemand, inventent la machine Enigma. Cette machine est la première forme de cryptage mécanique. Elle fut utilisée par les Allemands dès 1930 et durant la deuxième guerre mondiale.

Fonctionnement

- Regardons de plus près les différents composants.



Enigma avait un fonctionnement de cryptage simple. La machine comprend un clavier pour écrire le message, des roues pour le codage et enfin un panneau lumineux pour l'affichage du message codé.

Dans la machine trois roues de codage brouillent les lettres, et s'appellent donc *rotors brouilleurs*.

- Suivons le chemin parcouru par une lettre.

  Lorsqu'on presse une touche du clavier un courant electrique est envoyé dans la machine. Il passe tout d'abord par le tableau de connexion, qui permute éventuellement la lettre avec une autre. Ensuite le courant traverse les rotors et les fait tourner sur eux-mêmes : le premier rotor avance d'un cran. Dès qu'il a fait un tour complet (quand il a avancé de 26 crans), le deuxième avance d'un cran et ainsi de suite.

  Ce système a pour effet de modifier, à chaque pression d'une touche, la façon dont se réalise la substitution d'une lettre par une autre.

- Le courant passe ensuite dans le réflecteur qui renvoie la lettre à travers les rotors juqu'au tableau lumineux sur lequel s'affiche la nouvelle lettre, la lettre codée.

- Chaque matin il fallait décider de la configuration initiale de la machine :
  - Choisir les 6 paires de lettres qui seront échangées au niveau du tableau de connexions.
  - Placer les rotors dans un certain ordre sur l'axe (par exemple d'abord rotor 1 puis 3 ensuite 2 ou 2, 1, 3 ou...).
  - Choisir la position initiale de chaque rotor (chacun contient les 26 lettres de l'aphabet).

  Exemple de configuration :
  - Connections : A–F , G–M , J–B, D–K, R–Q, U–E.
  - Ordre des rotors : 1–3–2.
  - Position initiale des rotors : Z, J, S.

  Chaque jour, les allemands changaient la configuration initiale de la machine. Ils disposaient de petits carnets de bord avec les configurations initiales pour les deux mois suivants.

A l'attaque d'Enigma

Les Polonais sont les premiers à s'attaquer à Enigma. En 1940, ils emportent leurs connaissances en Angleterre. Les cryptanalystes anglais sont environ 3500 à travailler à Bletchley Park. Un des plus fameux d'entre eux est Alan TURING. Chaque jour l'équipe déchiffre de nombreux messages allemands. Durant toute la guerre, Bletchley Park est resté secret et les Allemands n'ont pas su que leurs messages étaient déchiffrés. Le cours de la guerre en a vraisemblablement été influencé.

5. De nos jours

Maintenant ce sont les ordinateurs qui cryptent les messages. Mais ce sont toujours des hommes qui inventent les algorithmes nécessaires, tant pour coder que pour décoder. Et la guerre des codes continue...

Pour en savoir plus

Histoire des codes secrets par Simon Singh, Ed. J.-C. Lattès

The Code Book on CD-Rom par Simon Singh and Nick Mee, Virtual Image and S. Singh

Les congruences de Gauss

Soient n, r et m des entiers. Si la différence $n - r$ est un multiple de m, on dit que « n est congru à r modulo m » et on note $n \equiv r \pmod{m}$.

Voici des exemples. 77 est congru à 17 modulo 12 puisque $77 - 17 = 60$ est divisible par 12. Un cas particulier important est celui de la division euclidienne : si r est le reste de la division de D par d, alors $D \equiv r \pmod{d}$. Ainsi $77 \equiv 5 \pmod{12}$. Sur une montre, nous sommes habitués à travailler modulo 12. A bien des égards les congruences se comportent comme des égalités.

Par exemple si $a \equiv b \pmod{m}$ et $c \equiv d \pmod{m}$ et si $\alpha \in \mathbb{N}$, alors

$$\begin{aligned} \alpha a &\equiv \alpha b \pmod{m} \\ a + \alpha &\equiv b + \alpha \pmod{m} \\ a + c &\equiv b + d \pmod{m} \\ a \cdot c &\equiv b \cdot d \pmod{m} \\ a^\alpha &\equiv b^\alpha \pmod{m} \end{aligned}$$

Exemples de calcul modulo 1073 :

$$\begin{aligned} 90^2 &= 8100 = 1073 \times 7 + 589 \equiv 589 \\ 90^4 &\equiv 589^2 = 346921 = 1073 \times 323 + 342 \equiv 342 \\ 90^8 &\equiv 342^2 = 116964 = 1073 \times 109 + 7 \equiv 7 \\ 90^{16} &\equiv 7^2 = 49 \\ 90^{23} &= 90 \times 90^2 \times 90^4 \times 90^{16} \\ &\equiv (90 \times 589) \times (342 \times 49) = 53010 \times 16758 \\ &\equiv 433 \times 663 = 287079 \equiv 588 \end{aligned}$$

Par contre, on ne peut pas toujours simplifier. Par exemple : $14 \equiv 2 \pmod{12}$ puisque 2 est le reste de la division de 14 par 12, mais $7 \not\equiv 1 \pmod{12}$.

Pour fabriquer un code nous aurons besoin d'*inverser un nombre w modulo m*. Il s'agit de trouver un entier $\bar{w}$ tel que $w \cdot \bar{w} \equiv 1 \pmod{m}$. Cela est toujours possible lorsque w et m sont premiers entre eux. De plus, un seul naturel compris entre 1 et m possède cette propriété.

Voici un exemple. Calculons l'inverse de 23 modulo 1008.

On commence par appliquer l'*algorithme d'Euclide* qui, par divisions successives permet de trouver le pgcd de deux nombres. Ceux-ci étant 23 et 1008, sont premiers entre eux et l'algorithme se termine par une division dont le reste vaut 1.

$$\begin{aligned} 1008 &= 23 \times 43 + 19 && (1) \\ 23 &= 19 \times 1 + 4 && (2) \\ 19 &= 4 \times 4 + 3 && (3) \\ 4 &= 3 \times 1 + 1 && (4) \end{aligned}$$

On remonte alors la chaîne de divisions, de façon à exprimer le nombre 1 sous la forme d'une *combinaison linéaire* de 23 et 1008.

$$\begin{aligned} 1 &\overset{(4)}{=} 4 - 3 \\ &\overset{(3)}{=} 4 - (19 - 4 \times 4) \\ &= 5 \times 4 - 19 \\ &\overset{(2)}{=} 5(23 - 19) - 19 \\ &= 5 \times 23 - 6 \times 19 \\ &\overset{(1)}{=} 5 \times 23 - 6 \times 1008 \\ &\quad +6 \times 43 \times 23 \\ &= 263 \times 23 - 6 \times 1008 \end{aligned}$$

On en déduit $1 \equiv 263 \times 23 \bmod 1008$. Ainsi, l'inverse de 23 modulo 1008 est 263.

Les nombres premiers

a) Un nombre premier est un nombre naturel qui possède exactement deux diviseurs distincts. Ainsi 2, 3, 5, 7, 11... sont premiers, 1 ne l'est pas.

b) Deux naturels a et b sont premiers entre eux si leur plus grand commun diviseur vaut 1. On dit aussi que a est premier avec b, ou à b.

c) Si un nombre premier divise le produit de deux naturels, il divise au moins un des deux facteurs.

d) Si un naturel est premier avec deux autres, il est premier avec leur produit.

Math-Jeunes n° 110S, 20–24, 2005

Un codage à clef révélée

Isabelle Pays

De tout temps l'homme a eu besoin de pouvoir transmettre des messages secrets. Actuellement, les transactions électroniques prennent de plus en plus d'ampleur (banques de données diverses sur les individus, renseignements médicaux, achats sur Internet...) Il faut pouvoir protéger les messages contre les indiscrétions ou les falsifications. Bref, il y a nécessité de recourir à des codages, c'est-à-dire des méthodes qui rendent l'information inintelligible à ceux à qui elle n'est pas destinée.
Bien sûr, les techniques utilisées en *cryptographie* (ensemble des techniques de codages) évoluent avec le temps. Des systèmes considérés comme inviolables il y a cinquante ans sont devenus aisés à déchiffrer avec la venue des ordinateurs. Les progrès incessants dans ce domaine remettent sans cesse en cause la sécurité des codages.
Les codages à clef révélée présentent la particularité qu'ils ne nécessitent pas l'échange d'informations secrètes entre l'émetteur et le destinataire du message.
Ils sont basés sur l'idée qu'il n'est pas possible de les déchiffrer « en un temps raisonnable » avec la capacité actuelle des ordinateurs, alors que pourtant la clef d'encodage est publique !

1. Des chiffres et des lettres

Pour appliquer les techniques de codage au message secret, on va d'abord le transformer en un message chiffré. Une méthode consiste à faire correspondre à chaque lettre (et signe de ponctuation), le numéro d'ordre correspondant dans l'alphabet.

A	B	C	...	Y	Z		.	!	?
0	1	2	...	24	25	26	27	28	29

Plus fréquemment, les gens travaillent avec le système binaire au lieu du système décimal. La table ci-dessous donne la traduction en binaire des numéros d'ordre de chaque lettre (et signe de ponctuation).

A	B	C	D	E	F	G	H	I	J
00000	00001	00010	00011	00100	00101	00110	00111	01000	01001
K	L	M	N	O	P	Q	R	S	T
01010	01011	01100	01101	01110	01111	10000	10001	10010	10011
U	V	W	X	Y	Z		.	!	?
10100	10101	10110	10111	11000	11001	11010	11011	11100	11101

2. Problème d'empilement

A) Etant donné des entiers a_1, a_2, ..., a_n et une somme S de certains de ces entiers, le *problème d'empilement* consiste à déterminer quels sont les entiers qu'il faut additionner pour obtenir la somme S. Une autre formulation du problème de l'empilement est de demander les valeurs x_1, x_2, ..., x_n, chacune étant 0 ou 1, pour que l'on ait

$$S = a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_nx_n \qquad (1)$$

Voici un exemple pour illustrer le problème d'empilement.

Soit $(a_1, a_2, a_3, a_4, a_5) = (2, 7, 8, 11, 12)$ et $S = 21$.

Par essais et erreurs, on trouve deux sous-ensembles d'entiers dont la somme est 21, à savoir $21 = 2 + 8 + 11 = 2 + 7 + 12$. L'autre point de vue (équivalent) nous dit qu'il y a deux solutions à l'équation

$$2x_1 + 7x_2 + 8x_3 + 11x_4 + 12x_5 = 21$$

avec $x_i = 0$ ou 1, à savoir $x_1 = x_3 = x_4 = 1$, $x_2 = x_5 = 0$ et $x_1 = x_2 = x_5 = 1$, $x_3 = x_4 = 0$.

Pour n et a_i suffisamment grands, même pour un ordinateur, le problème d'empilement est ardu à résoudre en un temps raisonnable.

B) Par contre, pour certaines valeurs de a_1, ..., a_n, la solution du problème d'empilement peut être beaucoup plus facile que dans le cas général. C'est le cas entre autres lorsque tout élément a_i (sauf le premier) est plus grand que la somme $a_1 + a_2 + \cdots + a_{i-1}$ des éléments qui le précèdent. De telles suites sont appelées « *super-croissantes* ».

Pour montrer que le problème d'empilement est facile à résoudre dans ce cas, nous allons considérer un exemple. Essayons de trouver quels entiers parmi $(2, 3, 7, 14, 27)$ ont leur somme égale à 37. D'abord, de $2 + 3 + 7 + 14 < 27$, on déduit que pour qu'une somme d'entiers de la suite $(2, 3, 7, 14, 27)$ soit plus grande que 27, il faut qu'elle contienne 27.

Donc, si $2x_1 + 3x_2 + 7x_3 + 14x_4 + 27x_5 = 37$ avec les $x_i = 0$ ou 1, on doit avoir $x_5 = 1$ et $2x_1 + 3x_2 + 7x_3 + 14x_4 = 10$. Puisque $14 > 10$, x_4 doit être 0 et l'équation devient : $2x_1 + 3x_2 + 7x_3 = 10$. Comme $2 + 3 < 7$, il faut prendre $x_3 = 1$. On a alors $2x_1 + 3x_2 = 3$ et clairement $x_2 = 1$ et $x_1 = 0$. La solution est donc $37 = 3 + 7 + 27$.

On imagine bien comment cela se généralise et on comprend donc bien que le problème d'empilement pour des suites $(a_1, \ldots, a_n)$ super-croissantes se résout très rapidement.

Une suite

$$(a_1, a_2, a_3, \ldots, a_{n-1}, a_n)$$

est super croissante si et seulement si quel que soit $i \geqslant 2$,

$$a_1 + a_2 + \cdots + a_{i-1} \leqslant a_i$$

Par exemple $(2, 3, 7, 14, 27)$ est super-croissante car
$2 < 3$, $2 + 3 < 7$,
$2 + 3 + 7 < 14$,
$2 + 3 + 7 + 14 < 27$.

Pour un nombre S donné, résoudre le problème d'empilement

$$S = a_1x_1 + \cdots + a_nx_n$$

avec $x_i = 0$ ou 1, lorsque $(a_1, a_2, \ldots, a_n)$ est super-croissante, c'est trouver successivement x_n, x_{n-1}, x_{n-2}, ..., x_1 par un raisonnement analogue à celui de l'exemple.

3. Application à la cryptographie à clef révélée

« A clef révélée » signifie que Monsieur « Tout-Le-Monde » a accès aux codes publics (dans un annuaire) et peut donc envoyer un message secret à n'importe quelle personne de l'annuaire. Par contre, seul le destinataire, sans jamais avoir échangé aucune sorte d'information particulière avec l'émetteur, est à même de déchiffrer le message.

Chaque individu (qui veut figurer à l'annuaire) choisit une suite d'entiers $(a_1, a_2, \ldots, a_n)$ super-croissante, pour laquelle le problème d'empilement est facile à résoudre. Ensuite il va fabriquer à partir de $(a_1, a_2, \ldots, a_n)$ une deuxième suite $(b_1, b_2, \ldots, b_n)$ pour laquelle le problème de l'empilement est « impossible » à résoudre en un temps raisonnable.

Pour cela il choisit un naturel m tel que $m > 2a_n$ et encore un naturel w premier à m. Il existe alors un et un seul naturel $\bar{w} < m$ tel que $\bar{w}.w \equiv 1 \pmod{m}$.

Lisez l'encadré consacré aux nombres premiers et aux congruences (page 19). Celles-ci constituent un outil indispensable pour coder !
Pour chaque congruence $a \equiv b \pmod{m}$ rencontrée dans le présent article, l'un des deux nombres a ou b est le reste de la division de l'autre par m (et est donc strictement inférieur à m).

Il fabrique la suite $(b_1, b_2, \ldots, b_n)$ en prenant pour b_j le reste de la division de $w \cdot a_j$ par m :

$$b_j \equiv w \cdot a_j \pmod{m} \text{ et } b_j < m$$

On obtient ainsi une suite « quelconque » $(b_1, b_2, \ldots, b_n)$ qui n'a plus aucune raison d'être super-croissante. Cette suite sera rendue *publique* dans un annuaire en face du nom de l'individu. (Comme un annuaire téléphonique.)

Annuaire : Monsieur X : $(b_1, \cdots, b_n)$

Le problème d'empilement

$$S = b_1 x_1 + \cdots + b_n x_n$$

où S est un entier donné, est donc difficile à résoudre.

Bien entendu, si $\bar{w}$, l'inverse modulo m de w, est connu, alors, en multipliant les deux membres par $\bar{w}$, on a :

$$\begin{aligned} \bar{w}S &= \bar{w} \cdot b_1 x_1 + \cdots + \bar{w} \cdot b_n x_n \\ &\equiv a_1 x_1 + \cdots + a_n x_n \pmod{m} \end{aligned}$$

puisque $\bar{w} \cdot b_j \equiv a_j \pmod{m}$. En posant S_0 le plus petit entier naturel congru à $\bar{w}S$ modulo m, on obtient le problème d'empilement :

$$S_0 = a_1 x_1 + \cdots + a_n x_n$$

qui est facile à résoudre (car la suite $(a_1, a_2, \ldots, a_n)$ est super-croissante), ce qui permet de trouver la solution $(x_1, x_2, \ldots, x_n)$ du problème initial.

Chaque individu garde secrets ses choix (de m, de w et des a_i). Chacun calcule son $\bar{w}$ (l'inverse de w modulo m) qu'il garde également secret.

Si quelqu'un souhaite envoyer un message secret à une personne de l'annuaire, il doit d'abord convertir son message lettré en un message chiffré ne contenant que des **0** et des **1**, comme décrit au point 2. Ensuite le message chiffré est coupé en blocs de longueur n. Si la longueur du message n'est pas divisible par n, on peut, par exemple, compléter le dernier bloc avec des **1**.

Pour chaque bloc, on calcule une somme S en utilisant la clef publique d'encodage qui figure dans l'annuaire à côté du nom du destinataire : si le bloc est $x_1 x_2 \ldots x_n$ (des **0** et des **1**), on obtiendra

$$S = b_1 x_1 + \cdots + b_n x_n$$

Les sommes S ainsi obtenues pour chaque bloc constituent le message codé qui est envoyé vers le destinataire.

Comme
$b_j \equiv w a_j \pmod{m}$
on a
$\bar{w} \cdot b_j \equiv \bar{w} \cdot w a_j \pmod{m}$
c'est-à-dire :
$\bar{w} \cdot b_j \equiv a_j \pmod{m}$
puisque
$\bar{w} \cdot w \equiv 1 \pmod{m}$

Si quelqu'un d'autre que le destinataire intercepte le message chiffré, il ne peut pas le déchiffrer « en un temps raisonnable » (problème d'empilement général). Par contre, le destinataire, qui seul connaît w et m, peut calculer $\bar{w}$ et donc se ramener à un problème d'empilement facile (comme on l'a vu auparavant).

Voici un exemple de codage-décodage. Je démarre avec la suite super-croissante :

$$(a_1, a_2, \ldots, a_{10}) = (2, 11, 14, 29, 58, 119, 241, 480, 959, 1917)$$

Je choisis $m = 3837$ (qui satisfait bien $m > 2a_{10}$) et $w = 1001$ (qui est bien premier à m). La suite transformée (par $b_j \equiv wa_j \pmod{m}$), $0 < b_j < m$) est

$$(b_1, b_2, \ldots, b_{10}) = (2002, 3337, 2503, 2170, 503, 172, 3347, 855, 709, 417).$$

C'est la clef publique, celle qui figure dans l'annuaire à côté de mon nom. Si quelqu'un veut m'envoyer le message suivant :

CODAGE

Il le traduit en binaire (voir table) puis le répartit en bloc de longueur 10 (c'est la longueur de ma suite de nombres dans l'annuaire).

$$\underbrace{\mathbf{00010}}_{C}\underbrace{\mathbf{01110}}_{O} \quad \mathbf{0001100000} \quad \mathbf{0011000100}$$

Pour chaque bloc, il calcule une somme en effectuant

$$b_1x_1 + \cdots + b_{10}x_{10} \text{ où } x_1x_2 \ldots x_{10} \text{ est un bloc.}$$

Par exemple, pour le premier bloc, on a $x_4 = x_7 = x_8 = x_9 = 1$ et $x_1 = x_2 = x_3 = x_5 = x_6 = x_{10} = 0$. La somme à calculer est donc $2170 + 3347 + 855 + 709$.

Mon correspondant obtient ainsi

7081 2673 5528

qui est le message codé qu'il m'envoie. Pour déchiffrer le message, j'utilise ma clef secrète : $\bar{w}$. Ici $\bar{w}$, l'inverse de 1001 modulo 3837, vaut 23 (car $23 \cdot 1001 = 23023 = 6 \cdot 3837 + 1 \equiv 1 \pmod{3837}$).

Pour déchiffrer le premier bloc, je calcule $7081 \cdot 23 \equiv 1709 \pmod{3837}$.

Ensuite, je résous le problème facile

$$2x_1+11x_2+14x_3+29x_4+58x_5+119x_6+241x_7+480x_8+959x_9+1917x_{10} = 1709$$

qui a pour solution

$$x_4 = x_7 = x_8 = x_9 = 1 \quad x_1 = x_2 = x_3 = x_5 = x_6 = x_{10} = 0$$

c'est-à-dire

$$(x_1x_2 \ldots x_{10}) = (\underbrace{\mathbf{00010}}_{C}\underbrace{\mathbf{01110}}_{O}) \quad \text{etc.}$$

Le principe du codage à clef révélée a été inventé par Whitfield Diffie, Ralph Merkle et Martin Hellman en 1976.

W. Diffie

M. Hellman

R. Merkle

Pour terminer, voici deux petits messages pour toi (fabriqués avec la même clef) :

Premier message
9586 12741 5027
2759 3801
Deuxième message
5414 10962 3384
8953 709 4619 3025
8400

4. Conclusions et remarques

1. Le système que je viens de décrire résout clairement deux des principaux problèmes des codages : la clef d'encodage est publique et la clef de décodage n'est connue que de celui qui va décoder. II n'y a donc plus le problème de transmettre, secrètement, la clef et un espion qui intercepte le message n'est pas capable de le déchiffrer en un temps raisonnable.
2. Il y a aussi moyen de résoudre le problème d'authentification avec ce système. Un autre article méritrait d'être consacré à ce sujet !
3. Vers 1980, Shamir a mis en évidence une faiblesse de ce système.
   Heureusement, différentes méthodes ont directement été mises au point pour éviter la faiblesse en question.
   Ensuite, d'autres systèmes à clef révélée ont vu le jour. Il y a eu par exemple en 1977 le fameux système RSA (nommé d'aprés de ses inventeurs, Rivest, Shamir, et Aldeman), qui est très résistant à l'ennemi.

En conclusion, on a vu comment les progrès dans le domaine des ordinateurs ont donné une nouvelle dimension aux codages. Cette branche continuera à évoluer aux rythmes des progrès aussi bien en informatique qu'en théorie des nombres ... Encore beaucoup de travail passionnant pour les générations futures...

Encoder et décoder un message à la main peut se révéler pénible. Peux-tu programmer un tableur (`Excel` par exemple) pour qu'il fasse les calculs à ta place ? Si oui, envoie-nous ta réalisation par e-mail, (adresse `sbpm@sbpm.be`) avec éventuellement un petit mode d'emploi. Soigne la présentation. Les meilleurs documents reçus seront placés sur le site Web de la SBPMef.

Solution du jeu n°1 : Des calculs non décimaux

Les calculs binaires

```
   1011            1011       1000 | 111
    101     ×      1101        111 | 1,001...
+  1101     ------------      ------
 ------            1011          1000
  11101          1011             111
                1011             -----
            ------------            1
               10001111
```

Les calculs hexadécimaux

×	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	2	4	6	8	A	C	E	10	12	14	16	18	1A	1C	1E
3	0	3	6	9	C	F	12	15	18	1B	1E	21	24	27	2A	2D
4	0	4	8	C	10	14	18	1C	20	24	28	2C	30	34	38	3C
5	0	5	A	F	14	19	1E	23	28	2D	32	37	3C	41	46	4B
6	0	6	C	12	18	1E	24	2A	30	36	3C	42	48	4E	54	5A
7	0	7	E	15	1C	23	2A	31	38	3F	46	4D	54	5B	62	69
8	0	8	10	18	20	28	30	38	40	48	50	58	60	68	70	78
9	0	9	12	1B	24	2D	36	3F	48	51	5A	63	6C	75	7E	87
A	0	A	14	1E	28	32	3C	46	50	5A	64	6E	78	82	8C	96
B	0	B	16	21	2C	37	42	4D	58	63	6E	79	84	8F	9A	A5
C	0	C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4
D	0	D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3
E	0	E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2
F	0	F	1E	2D	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1

```
   8A7B          8A7B
    187      ×   E520
+  E520      --------
 ------         114F6
  17122        2B467
              792BA
             --------
             7BF15660
```

Math-Jeunes n° 110S, 25–26, 2005

Le cryptosystème RSA

Françoise Valette

Nous exposerons ici le principe d'un seul système à clé révélée, apparemment très fiable et abondamment appliqué de nos jours : le système RSA. Il sert dans la plupart des communications sécurisées sur Internet, par exemple pour les paiements par carte de crédit ou lorsqu'une signature doit être authentifiée.
Il a été inventé en 1977 à l'Institut de Technologie du Massachusetts (USA) par Ronald Rivest, Adi Shamir et Leonard Aldeman.
Comme dans tout système à clé révélée, il s'agit de fabriquer un « coffre-fort mathématique » muni de deux clefs : l'une sert à le fermer, c'est-à-dire à encoder le message (clef E), et l'autre à l'ouvrir, c'est-à-dire à décoder le message (clef D). La clef E est rendue publique, tandis que la clef D n'est connue que par le destinataire.

L'indicateur d'Euler $\varphi(n)$ du naturel n est le nombre d'entiers premiers avec n qui appartiennent à $\{1, 2, \ldots n\}$.

$\varphi(1) = 1$, $\varphi(2) = 1$,
$\varphi(3) = 2$, $\varphi(4) = 2$,
$\varphi(5) = 4$, $\varphi(6) = 2$,
$\varphi(7) = 6 \ldots$

Si p et q sont deux nombres premiers, alors $\varphi(pq) = (p-1)(q-1)$

Fabrication du « coffre-fort » par le destinataire des messages

Exemple

Choix (au hasard) de deux « grands » (c'est-à-dire d'au moins 150 chiffres) nombres premiers distincts p et q. (Quelques minutes suffisent à un ordinateur.)	N'étant pas des ordinateurs, nous choisissons des nombres (trop) petits : $p = 29 \qquad q = 37$
Calcul de leur produit n (qui a au moins 300 chiffres)	$n = 29 \times 37 = 1073$
Calcul de l'indicateur d'Euler $\varphi(n)$ de n	$\varphi(n) = 28 \times 36 = 1008$
Choix d'un entier e entre 3 et n, et premier avec $\varphi(n)$.	$e = 23$ $(1008 = 2^4 \times 3^2 \times 7)$
Calcul de l'inverse d de e modulo $\varphi(n)$[1]	$d = 263$ car $263 \times 23 = 6 \times 1008 + 1 \equiv 1 \pmod{1008}$

e et n sont rendus publics ; p, q, $\varphi(n)$ et d sont secrets.

Verrouillage du coffre-fort (encodage du message)

Message $\mathcal{M}$: JE LIRAI MATH JEUNES

- Transcrire le message $\mathcal{M}$ à encoder sous forme numérique en utilisant la bijection ci-contre :

A	B	C	D	E	F	G	H	I	J	K	L	M
00	01	02	03	04	05	06	07	08	09	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

09 04 11 08 17 00 08 12 00 19 07 09 04 20 13 04 18

- Partitionner $\mathcal{M}$ en k blocs M_1, M_2, M_3, …, M_k d'égale longueur tels que chaque nombre M_i soit premier avec p et q. Dans la pratique, p et q ayant plus de 150 chiffres, il suffit que les blocs M_i aient moins de 150 chiffres chacun. (Si le dernier bloc est trop court, il est complété par le code du « blanc », ici 26).

M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}
090	411	081	700	081	200	190	709	042	013	041	826

- Encoder $\mathcal{M}$ en remplaçant chaque bloc M_i par le reste $E(M_i)$ de la division de M_i^e par n. Autrement dit, $E(M_i) \equiv M_i^e \pmod{n}$, avec $0 \leqslant E(M_i) < n$.

(1) La définition et les propriétés des congruences de Gauss, ainsi que le calcul de d, figurent à la page 19.

Par exemple (voir page 19) :

$$E(M_1) \equiv M_1^e = 90^{23} \equiv 588 \pmod{1073}$$
$$E(M_2) \equiv M_2^e = 411^{23} \equiv 062 \pmod{1073}$$

etc.

Le message codé à transmettre est donc **588 062...**

Décryptage

Considérons un bloc M et montrons que $(E(M))^d \equiv M \pmod{n}$.

$(E(M))^d \equiv (M^e)^d = M^{e\cdot d} \pmod{n}$

d est l'inverse de e modulo $\varphi(n)$. Si a est le quotient de $e \cdot d$ par $\varphi(n)$, on a

$$e \cdot d = a\varphi(n) + 1$$

Donc

$$(E(M))^d \equiv M \cdot M^{a\varphi(n)} \pmod{n}$$

Mais $\varphi(n) = (p-1)(q-1)$, donc

$$M^{a\varphi(n)} = M^{a(p-1)(q-1)}$$

Puisque q est premier et ne divise pas M, il ne divise pas non plus $M^{a(p-1)}$. Par le petit théorème de Fermat, $(M^{a(p-1)})^{q-1} - 1$ est alors divisible par q.

De même, $(M^{a(q-1)})^{p-1} - 1$ est divisible par p.

Donc $M^{a(p-1)(q-1)} - 1$ est divisible par $pq = n$.

Ainsi $M^{a(p-1)(q-1)} \equiv 1 \pmod{n}$ et $(E(M))^d \equiv M \pmod{n}$.

Ouverture du coffre-fort (décryptage du message)

Le destinataire est le seul à connaître d. Il reçoit la suite $E(M_1)\ \ E(M_2)\ldots$ Pour retrouver chaque M_i, il utilise la formule $(E(M_i))^d \equiv M_i \pmod{n}$. Par exemple $588^{263} \equiv 90 \pmod{1073}$. (Pour une preuve de la formule, voir ci-contre.) La clé de décodage est donc donnée par $D(E(M_i)) \equiv (E(M_i))^d \pmod{n}$, avec $0 \leqslant D(E(M_i)) < n$.

Signature d'un message

Supposons qu'une personne a souhaite envoyer un message $\mathcal{M}$ à une personne b et le signer (y mettre un signe d'authentification). Comment va-t-elle procéder ?

a applique à $\mathcal{M}$ sa clef (secrète) de décodage D_a, puis la clef (publique) de codage E_b du destinataire. b reçoit donc $E_b(D_a(\mathcal{M}))$. Pour décrypter, il lui suffit d'appliquer D_b puis E_a.

Si le message décodé a un sens, b est certain que a en est l'expéditeur, puisque seul a connaît la clef D_a.

Sécurité de la « serrure »

Tous les algorithmes imaginés jusqu'ici pour découvrir la clef D, c'est-à-dire calculer d connaissant e et n, se ramènent à la factorisation de n en $p \times q$. Suivant la puissance des machines on aura besoin d'un temps variant de quelques dizaines d'années à quelques milliards de milliards d'années (!) pour casser un code dont la clef comporte 300 chiffres, ce qui est par conséquent tout à fait déraisonnable et inutilisable.

Pouvons-nous conclure qu'un système RSA est inviolable ? Peut-être écrira-t-on un jour un algorithme permettant de calculer d en un temps raisonnable, ou même de restituer le texte en clair sans connaître d. À moins que l'ordinateur quantique ne change fondamentalement les données du problème. Voyez à ce sujet la note de la page 14.

Petit théorème de Fermat

Si p est un nombre premier et n un naturel non divisible par p, alors

$$n^{p-1} \equiv 1 \pmod{p}$$

Autrement dit, $n^{p-1} - 1$ est divisible par p.

Exemples

$3^4 = 81 \equiv 1 \pmod{5}$

$6^2 = 36 \not\equiv 1 \pmod{3}$

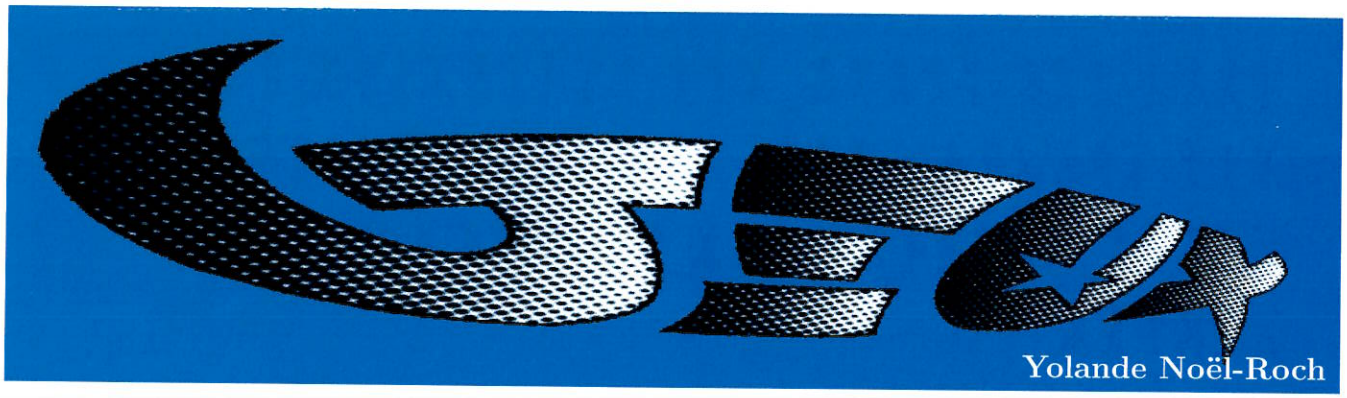
$12^6 = 144^3 \equiv 4^3 = 64 \equiv 1 \pmod{7}$

Les inventeurs du système RSA

Ronald Rivest

Adi Shamir

Leonard Aldeman



Solutions des jeux, page 24 et page 3 de couverture.

1. Des calculs non décimaux

Effectuez les opérations suivantes en binaire :

$$\begin{array}{r} 1011 \\ 101 \\ +\ 1101 \\ \hline \end{array} \qquad \begin{array}{r} 1011 \\ \times\ 1101 \\ \hline \end{array} \qquad 1000 \,\big|\, 111$$

Complétez la table de multiplication hexadécimale :

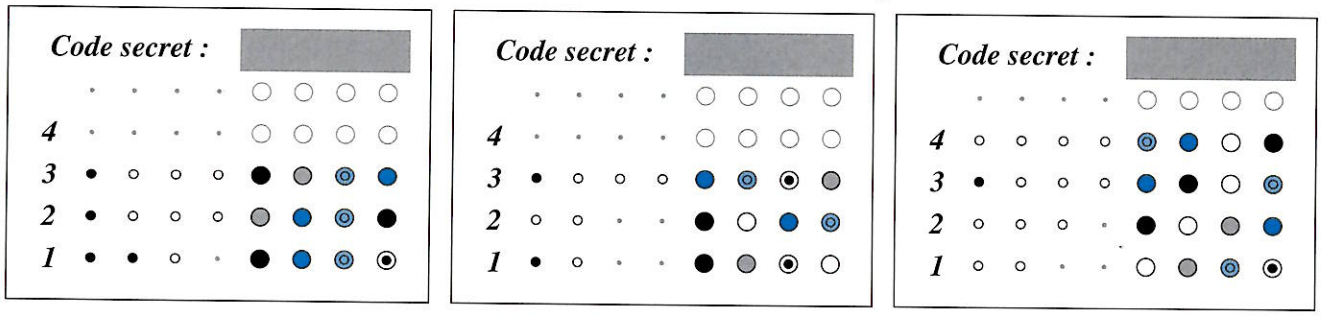
×	0	1	2	3	4	5	6	7	8	9	*A*	*B*	*C*	*D*	*E*	*F*
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	*A*	*B*	*C*	*D*	*E*	*F*
2	0	2	4	6	8	*A*	*C*	*E*	10	12	14	16	18	1*A*	1*C*	1*E*
3	0	3	6	9	*C*	*F*	12	15	18	1*B*	1*E*	21	24	27	2*A*	2*D*
4	0	4	8	*C*	10	14	18	1*C*	20	24	28	2*C*	30	34	38	3*C*
5	0	5	*A*	*F*	14	19	1*E*	23	28	2*D*	32	37	3*C*	41	46	4*B*
6	0	6	*C*	12	18	1*E*	24	2*A*	30	36	3*C*	42	48	4*E*	54	5*A*
7	0	7	*E*	15	1*C*	23	2*A*	31	38	3*F*	46	4*D*	54	5*B*	62	69
8	0	8	10	18	20	28	30	38	40							
9	0	9	12	1*B*	24	2*D*	36	3*F*	48							
A	0	*A*	14	1*E*	28	32	3*C*	46	50							
B	0	*B*	16	21	2*C*	37	42	4*D*	58							
C	0	*C*	18	24	30	3*C*	48	54	60							
D	0	*D*	1*A*	27	34	41	4*E*	5*B*	68							
E	0	*E*	1*C*	2*A*	38	46	54	62	70							
F	0	*F*	1*E*	2*D*	3*C*	4*B*	5*A*	69	78							

Effectuez :

$$\begin{array}{r} 8\,A\,7\,B \\ 1\,8\,7 \\ +\ E\,5\,2\,0 \\ \hline \end{array} \qquad \begin{array}{r} 8\,A\,7\,B \\ \times\ E\,5\,2\,0 \\ \hline \end{array}$$

2. Jouez au Master Mind

Dans les parties suivantes de Master Mind, les quatre pions cachés sont de couleurs différentes. Les parties sont inachevées. Pouvez-vous les terminer en un seul coup ?

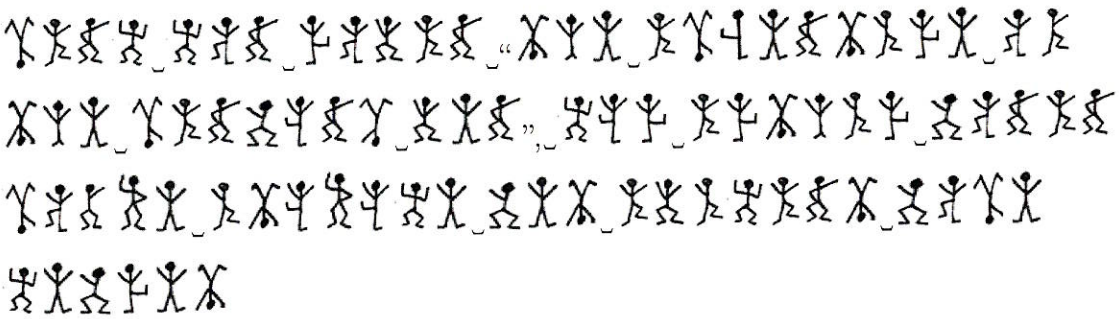


3. Des messages à décoder

Message n°1 :

UHFRQQDLVVHC-YRXV OH FRGH VHFUHW GH MXOHV FHVDU ?

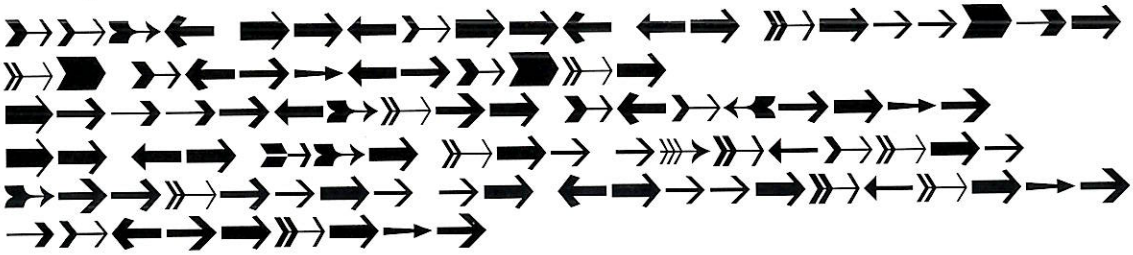
Message n°2



Message n°3

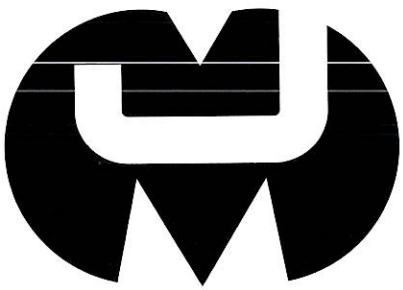
YX UDLMVWRJ JGAXDL KX AWFV-ZXGJXI WGDW ULGD FVXAX
"YXI BXLAXFDRXI"

Message n°4

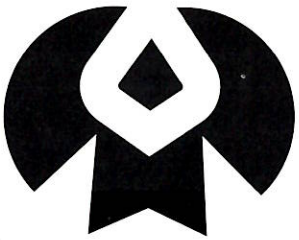


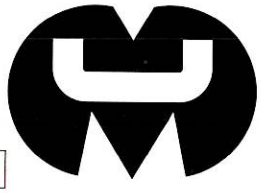
4. Des symétries « miroir »

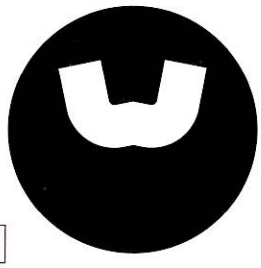
Placez le miroir fourni dans le numéro 109 de *Math-Jeunes* sur la figure ci-dessous de manière à créer les figures suivantes.

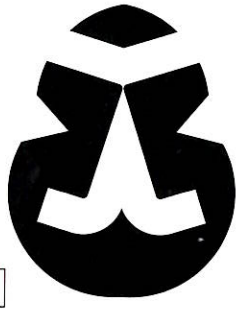


Ces dessins peuvent être associés en paires complémentaires : deux dessins complémentaires sont obtenus en plaçant le miroir au même emplacement sur l'image précédente, mais en inversant les deux faces du miroir. Les deux dessins d'une paire ne sont pas nécessairement à la même échelle, ni avec la même orientation. Pouvez-vous reconstituer les associations ?


1


2

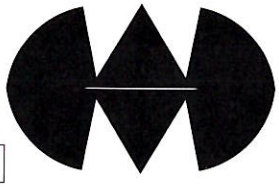

3

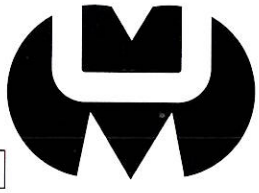

4

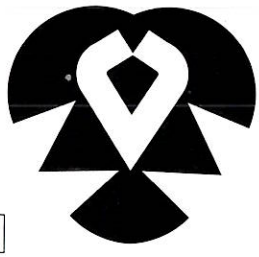

5

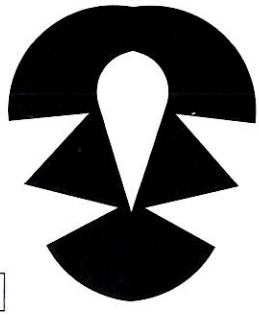

6

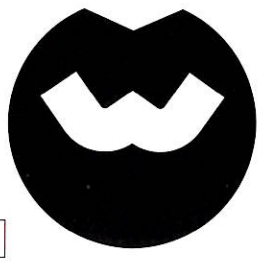

7

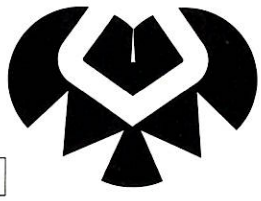

8


9


10


11

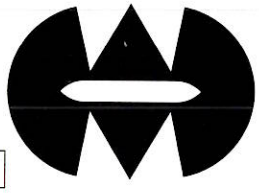

12


13

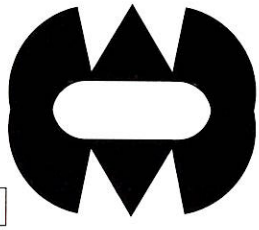

14

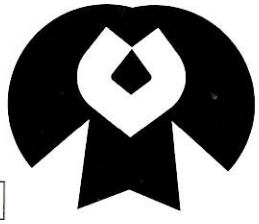

15


16


17


18


19


20


21


22

Olympiades

Claudine Festraets

Tu viens certainement de participer à l'éliminatoire de l'Olympiade Mathématique Belge. Tu seras peut-être admis en demi-finale, et dans ce cas, je te félicite, mais si tu n'as pas cette chance, ne t'en fais pas, prends courage et réinscris-toi l'année prochaine. Dans tous les cas, voici quelques énoncés qui devraient t'intéresser ; ces énoncés sont choisis parmi ceux des épreuves midi et maxi de cette dernière éliminatoire, essaye d'abord de les résoudre sans regarder la solution.

Si tu désires t'exercer davantage, tu peux te procurer les tomes 4 et 5 des brochures « Olympiades ». Voici tous les renseignements nécessaires pour les commander.

Olympiades Mathématiques Belges
Tome 4 (1994-1998) : 5 €
Tome 5 (1999-2002) : 6 €
Tome 4 et tome 5 : 10 €

Les commandes sont à adresser à
SBPMef, rue de la Halle, 15, 7000 Mons
Compte : 000-0728014-29
Fax et téléphone : 065 37 37 29.

Ajouter 1,80 € de port pour un tome et 3,50 € de port pour deux tomes.

Midi 9 – Maxi 5

Combien de diviseurs positifs de 9^9 sont des cubes parfaits ?

Ⓐ 6 Ⓑ 7 Ⓒ 8 Ⓓ 9 Ⓔ 18

Solution

On a $9^9 = \left(9^3\right)^3 = 729^3$. Le nombre de diviseurs de 9^9 qui sont des cubes parfaits est égal au nombre de diviseurs de 729. Or l'ensemble des diviseurs de 729 est {1,3,9,27,81,243,729}. Le nombre cherché est donc 7.

Midi 17

Les nombres réels a et b vérifient l'égalité $\frac{a}{b} = \frac{b}{a}$. Dans ce cas, a et b sont nécessairement tels que

Ⓐ $a = b = 1$ Ⓑ $a = b$ Ⓒ $ab = 1$

Ⓓ $|a| = |b|$ Ⓔ $a + b = ab$

Solution

L'égalité $\frac{a}{b} = \frac{b}{a}$ est vérifiée par les réels a et b, donc a et b sont non nuls et l'égalité est équivalente à $a^2 = b^2$. Si a et b sont de même signe, alors $a^2 = b^2 \Leftrightarrow a = b$; mais si a et b sont de signes opposés, alors $a^2 = b^2 \Leftrightarrow a = -b$. Dans l'un et l'autre cas, on obtient $|a| = |b|$.

Midi 22 – Maxi 10

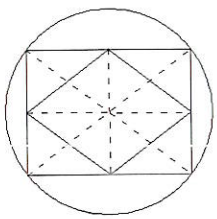
Un rectangle est inscrit dans un cercle de rayon R. Quelle est la longueur du côté du losange qui a pour sommets les milieux des côtés de ce rectangle ?

Ⓐ $\sqrt{R^2 - 1}$ Ⓑ $\sqrt{R^2 + 1}$ Ⓒ R Ⓓ $2R\frac{\sqrt{3}}{3}$

Ⓔ Cette longueur dépend des dimensions du rectangle.

Solution

Les diagonales du rectangle et les diagonales du losange se coupent au centre du cercle. Les diagonales du losange partagent le rectangle en 4 petits rectangles et dans chacun de ceux-ci, les diagonales ont même longueur.



Une des diagonales d'un petit rectangle est un côté du losange et l'autre est le rayon du cercle, donc la longueur du côté du losange est R.

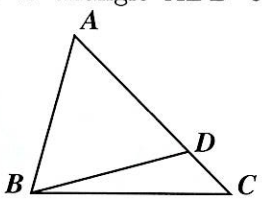
Midi 24

Dans le triangle ABC, nous avons $\widehat{ABC} - \widehat{ACB} = 30°$. Un point D situé sur $[AC]$ est tel que $|AD| = |AB|$. La mesure en degrés de l'angle $\widehat{CBD}$ est

Ⓐ 12 Ⓑ 15 Ⓒ 17 Ⓓ 21 Ⓔ 23

Solution

Nous utiliserons le fait que le triangle ABD est isocèle et que, dans un triangle, un angle extérieur est égal à la somme des deux angles intérieurs non adjacents.



$$\begin{aligned} 30° &= \widehat{ABC} - \widehat{ACB} \\ &= \widehat{ABD} + \widehat{DBC} - \widehat{ACB} \\ &= \widehat{DBC} + \widehat{DCB} + \widehat{DBC} - \widehat{ACB} \\ &= 2\widehat{DBC} \end{aligned}$$

d'où $\widehat{DBC} = 15°$.

Midi 27

Les nombres entiers a, b, c sont donnés par

$$a = 2^{5555}, \quad b = 3^{3333}, \quad c = 6^{2222}$$

Laquelle des doubles inégalités ci-dessous est vraie ?

Ⓐ $a < c < b$ Ⓑ $a < c < b$ Ⓒ $c < b < a$

Ⓓ $b < c < a$ Ⓔ $b < a < c$

Solution

$a = 2^{5555} = \left(2^5\right)^{1111} = 32^{1111}$, $b = 3^{3333} = \left(3^3\right)^{1111} = 27^{1111}$, $c = 6^{2222} = \left(6^2\right)^{1111} = 36^{1111}$ Ce qui donne $b < a < c$.

Midi 30 – Maxi 11

Si x maçons mettent y jours pour bâtir z maisons, combien de jours mettront q maçons pour bâtir r maisons (en supposant que tous les maçons travaillent au même rythme et que toutes les maisons sont identiques) ?

Ⓐ $\frac{qry}{xz}$ Ⓑ $\frac{ryz}{qx}$ Ⓒ $\frac{qz}{rxy}$ Ⓓ $\frac{xyr}{qz}$ Ⓔ $\frac{rz}{qxy}$

Solution

Un maçon met yx jours pour bâtir z maisons et $\frac{yx}{z}$ jours pour bâtir une maison. Donc q maçons mettent $\frac{yx}{qz}$ jours pour bâtir une maison et $\frac{yxr}{qz}$ jours pour bâtir r maisons.

Maxi 15 Le nombre réel positif non nul x est tel que $\left(x + \frac{1}{x}\right)^2 = 5$. Que vaut $x^3 + \frac{1}{x^3}$?

Ⓐ 0 Ⓑ 1 Ⓒ $2\sqrt{5}$ Ⓓ $4\sqrt{5}$ Ⓔ 10

Solution

$$\begin{aligned} x^3 + \frac{1}{x^3} &= x^3 + 3x^2\frac{1}{x} + 3x\frac{1}{x^2} + \frac{1}{x^3} - 3x^2\frac{1}{x} - 3x\frac{1}{x^2} \\ &= \left(x + \frac{1}{x}\right)^3 - 3x^2\frac{1}{x} - 3x\frac{1}{x^2} \\ &= (\sqrt{5})^3 - 3x - 3\frac{1}{x} \\ &= 5\sqrt{5} - 3\sqrt{5} = 2\sqrt{5} \end{aligned}$$

Maxi 21 Dans le plan, la courbe d'équation $y = x^3 + 3x^2 + 3x + 5$

Ⓐ n'admet aucun centre de symétrie ;

Ⓑ admet un centre de symétrie de coordonnées $(0, 5)$;

Ⓒ admet un centre de symétrie de coordonnées $(1, -4)$;

Ⓓ admet un centre de symétrie de coordonnées $(-1, 4)$;

Ⓔ admet un centre de symétrie de coordonnées $(-1, -4)$.

Solution

L'équation donnée peut s'écrire $y = (x + 1)^3 + 4$ ou encore $y - 4 = (x + 1)^3$. La courbe d'équation $y - 4 = (x + 1)^3$ est l'image de la courbe d'équation $y = x^3$ par la translation qui applique le point $(0, 0)$ sur le point $(-1, 4)$. Or la courbe d'équation $y = x^3$ est symétrique par rapport au point $(0, 0)$, donc le centre de symétrie de la courbe d'équation $y - 4 = (x + 1)^3$ est le point $(-1, 4)$.

Maxi 23 Arthur construit une fonction f telle que $f(f(x)) = f(x + 2) - 3$ pour tout entier x, avec de plus $f(1) = 4$ et $f(4) = 3$. Dans ces conditions, $f(5)$

Ⓐ vaut 3 ; Ⓑ vaut 6 ; Ⓒ vaut 9 ; Ⓓ vaut 12 ; Ⓔ ne peut être déterminé.

Solution

$f(f(1)) = f(1 + 2) - 3$ et $f(1) = 4$, d'où $f(4) = f(3) - 3$, mais $f(4) = 3$, d'où $f(3) = 6$. $f(f(4)) = f(4 + 2) - 3$ et $f(4) = 3$, d'où $f(3) = f(6) - 3$, mais $f(3) = 6$, d'où $f(6) = 9$. $f(f(3)) = f(3 + 2) - 3$ et $f(3) = 6$; d'où $f(6) = f(5) - 3$, mais $f(6) = 9$, d'où $f(5) = 12$.

Maxi 29 Un sac contient quatre balles portant les numéros -2, -1, 1 et 2. On tire simultanément deux balles du sac. Quelle est la probabilité que le produit de leurs numéros soit positif et pair ?

Ⓐ $\frac{1}{6}$ Ⓑ $\frac{1}{2}$ Ⓒ $\frac{1}{3}$ Ⓓ $\frac{2}{3}$ Ⓔ $\frac{3}{7}$

Solution

Examinons tous les tirages possibles : -2 et -1, le produit vaut 2 ; -2 et 1, le produit vaut -2 ; -2 et 2, le produit vaut -4 ; -1 et 1, le produit vaut -1 ; -1 et 2, le produit vaut -2 ; 1 et 2, le produit vaut 2. Le produit est positif et pair dans deux cas sur six, la probabilité demandée est donc $\frac{1}{3}$.

RALLYE problèmes

Nicole Miewis

Vous trouverez ci-dessous les cinq problèmes de la seconde étape. Envoyez vos solutions à Nicole Miéwis, Avenue de Péville, 150, 4030 - Grivegnée, munies de la mention « Rallye Math-Jeunes » avant le 20 mars 2005. Les solutions les plus élégantes seront publiées dans *Math-Jeunes*.

6. Le cochonnet

(3 points)

Trois boules de pétanque et un cochonnet, sont tous posés sur une table ; de plus, chacune des quatres sphères est tangente aux trois autres.

Quel est le rayon du cochonnet si les boules ont un diamètre de 9 cm ?

7. Le tableau

(3 points)

On dispose d'un tableau de 4 lignes, numérotées de 0 à 3 et de 2005 colonnes, numérotées de 0 à 2004. À l'intersection de la ligne m et de la colonne n se trouve l'élément noté $f(m,n)$. On donne, pour tout entier compris entre 0 et 2004 la valeur

$$f(0,n) = n+1,$$

ce qui permet de remplir la première ligne du tableau.

Par ailleurs, on sait que pour tout entier m compris entre 1 et 3 et pour tout entier n compris entre 1 et 2004, on a les égalités suivantes :

$$f(m,0) = f(m-1,1)$$

et

$$f(m,n) = f(m-1, f(m,n-1)).$$

Quel nombre se trouve à l'intersection de la dernière ligne et de la dernière colonne ?

8. Les calissons

(10 points)

Un *calisson* est une confiserie à base d'amande recouverte de sucre glacé, spécialité d'Aix-en-Provence. Ces bonbons ont la forme d'un prisme droit de hauteur y, dont la base est un *losange* de côté z obtenu par juxtaposition de deux triangles équilatéraux.

Un confiseur souhaite ranger 216 calissons bien serrés sur 8 couches, ne laissant entre elles qu'un espace négligeable, dans une boîte de 0,972 dm^3, ayant la forme d'un prisme droit dont les bases sont des hexagones réguliers de côté x. Les calissons seront posés de manière à ce que leurs bases soient parallèles aux bases de la boîte.

Si la boîte est construite pour que son aire totale soit minimale, quelles sont, au mm près, les longueurs du côté x des bases de la boîte, du côté z et de la hauteur y d'un calisson ?

9. Drôle de nombre

(7 points)

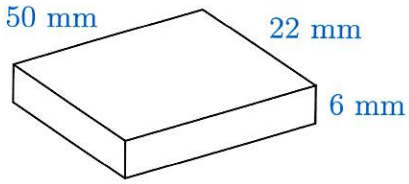
Ce nombre est formé des chiffres 1 à 9 qui y figurent une fois chacun. Le nombre formé par les 2 premiers chiffres est divisible par 2, le nombre formé par les 3 premiers chiffres est divisible par 3, ... le nombre formé par les 9 premiers chiffres est divisible par 9.

Quel serait un de ces drôles de nombres ?

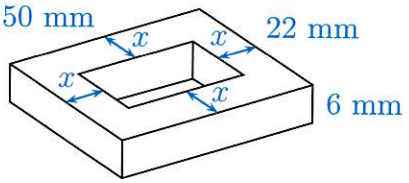
10. Les gaufrettes

(5 points)

Une gaufrette se compose d'un biscuit recouvert de chocolat. La forme du biscuit peut être assimilée à un parallélépipède rectangle dont les dimensions sont 50 mm, 22 mm et 6 mm. La couche de chocolat étant très fine, on convient d'en négliger l'épaisseur. Ainsi donc, le volume de chocolat qui recouvre la gaufrette s'exprimera par le même nombre que l'aire de la surface sur laquelle repose le chocolat.



Le fabricant souhaite réaliser dans ce biscuit un trou, également de forme parallélépipèdique. Ce trou sera percé dans les plus grandes faces du biscuit, parallèlement aux arêtes de la gaufrette. Ses parois seront recouvertes de la fine couche de chocolat. Le trou doit être situé à égale distance des deux bords du biscuit. On appelle x cette distance.



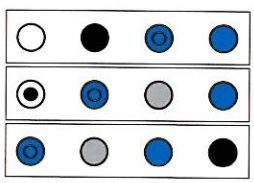
Le fabricant se dit qu'il existe une valeur de x pour laquelle il utiliserait *la même quantité de chocolat pour la gaufrette de départ et pour la gaufrette trouée.*

Quelle est cette valeur de x ?

Solutions des jeux n°2 à 4

2. Jouez au Master Mind

Combinaisons à découvrir :



3. Des messages à décoder

Message n°1 :

RECONNAISSEZ-VOUS LE CODE SECRET DE JULES CESAR ?

Message n°2 :

DANS SON ROMAN "THE ADVENTURE OF THE DANCING MEN", SIR ARTHUR CONAN DOYLE UTILISE CET AMUSANT CODE SECRET

Message n°3 :

LE PROCHAIN NUMERO DE MATH-JEUNES AURA POUR THEME "LES GEOMETRIES"

Message n°4 :

POUR DECODER CE MESSAGE LA PRINCIPALE DIFFICULTE PROVIENT DE CE QUE LES SYMBOLES UTILISES SE RESSEMBLENT FORTEMENT

4. Des symétries « miroir »

Les dessins associés sont : 1 et 5, 2 et 9, 3 et 16, 4 et 11, 6 et 20, 7 et 22, 8 et 15, 10 et 21, 12 et 13, 14 et 17, 18 et 19.

Math-Jeunes
Périodique trimestriel
15, rue de la Halle – 7000 Mons
Bureau de dépôt 7000 Mons 1

Responsable de l'édition: G . NOËL
Rue de la Culée 86 - 6927 Resteigne
Bureau de dépôt : Mons 1

Autorisation de fermeture
Sluitings toelating

7000 Mons 1
5/156

Belgique - België
P.P.
7000 Mons 1
5/124

Périodique - Peut être ouvert pour contrôle postal

Réservé à la poste

Inconnu
Refusé
Décédé
Adresse insuffisante
N'habite plus à l'adresse indiquée