UNE BRÈVE HISTOIRE DES NOMBRES

PHILIPPE LANGEVIN

Notes des cours d'algorithmique arithmétique 1999-2000 pour les étudiants du I31 de la faculté des sciences et techniques de Toulon et de la licence de mathématiques de l'université Gaston Berger à Saint-Louis au Sénégal. Version de Septembre 2000, corrigée en Novembre 2003.

Table des matières

1. Présentation des acteurs

La petite histoire des nombres que je vais vous présenter prend place dans la grande histoire des mathématiques, et des sciences en général. Les découvertes archéologiques montrent que l'histoire (occidentale) des nombres prend source chez les égyptiens et les babyloniens. Elle se développe sous la Grèce antique, puis s'exile en Perse. De retour en Europe au moyen age, elle renaît de ses cendres au cours du siècle des lumières. Voilà

Date: Octobre 1999, Septembre 2000.

en quelques mots, d'où nous vient *l'arithmétique*. Très vite les nombres sont utilisés pour leurs aspects comptables, mais les « propriétés cachées » des nombres, fondamentales et profondes restent sans applications concrètes.

> *...both Gauss and lesser mathematicians may be justified in rejoicing that there is one science (number theory) at any rate, and that their own, whose remoteness from ordinary human activities should keep it gentle and clean.*
>
> –G. H. HARDY, A mathematician'Apology, 1940

Il encore souhaitable de penser comme le mathématicien Hardy mais les années soixante-dix vont changer le statut de l'arithmétique. En effet, toutes les connaissances théoriques accumulées au cours du temps, sur les nombres et les « indomptables » nombres premiers, débouchent sur une nouvelle génération de cryptosystèmes du plus simple : RSA (objet de cette note), aux plus complexes qui sont basés sur les courbes elliptiques et les variétés abéliennes. Sachez le de suite, l'utilisation de ces méthodes de chiffrement tend à devenir quotidienne pour la plupart d'entre nous! L'utilisateur d'un système `unix` doit montrer patte blanche en soumettant un nom de login et un mot de passe. Tous les mots de passes sont cryptés et enregistrés dans le fichier `/etc/password`.

```
benza:gHedHLTOZ7pNs:13:0:Didier Benza,127R,33494142292,:/home/benza:/bin/t
httpd:cEGcZQS0JTnZU:14:31:httpd:/usr/local/apache:/bin/tcsh
langevin:gRYCljlfjECcE:272:41:Philippe Langevin,:/home/Gect/langevin:/bin/
zanotti:tlxHS7tFsiWj2:273:41:Jean-Pierre ZANOTTI,:/home/Gect/zanotti:/bin/
veron:50o6dyzrqCiP2:282:41:Pascal Veron:/home/Gect/veron:/bin/tcsh
gontard:PGbt01m43POdI:283:41:Christine Gontard:/home/Gect/gontard:/bin/don
jld:yiJZ.75Vb6zbM:284:41:Jean-Luc Damoiseaux,:/home/Gect/jld:/bin/tcsh
muri:ry4xuNuEvHvj.:288:41:Elisabeth Murisasco:/home/Gect/muri:/bin/csh
gillot:lWfRsA9zpkO82:295:41:Gillot Valerie:/home/Gect/gillot:/bin/dontlogi
nguyen:ECRlY08jOxA1Y:296:41:Nguyen Christian,:/home/Gect/nguyen:/bin/tcsh
lemaitre:JaYzOcY6X3wvk:297:41:Lemaitre Jacques:/home/Gect/lemaitre:/bin/tc
rabizzon:ijrusepcVUgQM:302:41:Patrice Rabizzoni:/home/Gect/rabizzon:/bin/d
wolfmann:dp7Pb6A2MqPHE:347:41:Jacques Wolfmann:/home/Gect/wolfmann:/bin/tc
```

L'objet de cette note est de mettre en évidence les grandes étapes qui sont à l'origine du RSA, un algorithme de chiffrement à clefs publiques inventé par Rivest, Shamir et Adleman. Avant tout, commençons par présenter quelques uns des principaux acteurs. Ils sont nombreux, plusieurs centaines, alors pour faire court, j'en ai sélectionné dix-sept.

Pythagore de Samos Philosophe du VI-ième siècle avant Jésus-Christ, Pythagore est l'un des inventeurs des mathématiques pures. Pour l'école Pythagoricienne qui explore les nombres en quête de vérité quasi-religieuse, les nombres sont la source et le principe de toute chose.

Phidias Sculpteur contemporain de Pythagore, Phidias est inspiré par son époque. Il propose des règles d'esthétiques fondées sur des concepts mathématiques à l'image du nombre $\phi = \frac{1+\sqrt{5}}{2} = 1,61803\ldots$, c'est le nombre d'or, la divine proportion d'Euclide.

Euclide d'Alexandrie Né autour du II-ième siècle avant Jésus-Christ. Son oeuvre *Les Éléments* est composée de treize livres, neuf d'entre-eux sont consacrés à la géométrie,

les quatre autres traitent des nombres et de l'arithmétique. Notamment le livre sept, dans lequel il expose une méthode pour calculer le *plus grand diviseur commun* de deux nombres entiers.

Diophante d'Alexandrie La liste des questions qu'il propose dans son traité historique *arithmétiques* marque le début d'une discipline. Mal situé dans le temps, entre -150 et +350 ! Il semble qu'une partie de son oeuvre se soit égarée. Les textes de Diophante seront remis au goût du jour par Bachet puis Fermat.

al-Kharezsmi Abu Ja'far Mohammed ibn Musas al-Kwarizmi. Né à Bagdad, vers 780 après Jésus-Christ. Il est l'auteur du traité *Kitab al'jabr w'al-muqabala* ce qui signifie *règles de restaurations et réductions*, un titre qui permet d'identifier l'un des inventeurs de l'algèbre. Par ailleurs, il semble que le mot *algorithme* soit une déformation de *al-Kharezsmi*.

Léonard de Pise dit « Fibonacci ». Probablement natif de Pise en 1170. Il fait ses études en Afrique du nord. Lecteur d'al-Kharezmi, il pourraît à l'origine du mot algorithme, et de l'introduction des chiffres arabes en Europe. Fibonacci est considéré l'arithméticien majeur pour la période post-Diophante à Fermat. La suite et les nombres de Fibonacci jouent un rôle déterminant dans cette note.

Claude Gaspard Bachet de Méziriac Né le 9 Octobre 1581 à Bourg-en-Bresse, auteur de nombreux casse-têtes mathématiques, il découvre une méthode de construction de carrés magiques. Il est le traducteur des *arithmétiques* de Diophante. Enfin, d'après Jean Itard, c'est à Bachet qu'il faut attribuer l'identité de Bézout.

Pierre de Fermat Natif de Toulouse en 1601, la lecture de Diophante par Pierre de Fermat relance l'intérêt des nombres. Le *grand théorème* de Fermat suscite vocations et passions pour des générations de mathématiciens.

Léonhard Euler Né à Bâle en 1707. Ici, nous découvrirons et utiliserons une infinitésimale partie de son oeuvre colossale : 5 mètres de rayonnage de la section « oeuvres » de la bibliothèque du CIRM, le Centre de International de Rencontres Mathématique de Luminy !

Gabriel Lamé Né à Tours en 1795. Étudiant puis professeur à l'école polytechnique. Bien connu des mécaniciens, il travaille dans des domaines nombreux et variés. En théorie des nombres, il prouve le cas $n = 7$ de la conjecture de Fermat. Acteur essentiel de notre pièce, il détermine la *complexité* de l'algorithme d'Euclide.

Carl Friederich Gauss Très difficile de parler de nombres sans parler du grand Carl Gauss, le prince des mathématiques. Natif de Brunswick (1777), il est considéré comme le meilleur arithméticien de tous les temps. Fidèle à sa devise :« peu mais mûr ! », Gauss écrit peu mais bien et c'est au travers de ses *disquisitiones arithmeticae*, il nous communique son enthousiasme vis-à-vis des nombres et de leurs « propriétés cachées ».

Bernhard Riemann Né à Breselen en 1826, Riemann est célèbre pour ses travaux en analyse complexe, géométrie analytique, intégration et topologie. Nous retenons de ses travaux d'une grande profondeur et d'une grande originalité l'extraordinaire *hypothèse de Riemann* intrinsèquement liée aux mystérieux nombres premiers.

Édouard Lucas Né le 4 avril 1842 à Amiens. Il travaille sous la direction de Le Verrier à l'observatoire de Paris. Auteur des *récréations mathématiques*, il est à l'origine du puzzle

des « *tours de Hanoï* » souvent choisi pour illustrer les mécanismes récursifs. Par ailleurs, il détient le record du plus grand nombre premier découvert sans l'aide d'une machine ; Il s'agit du nombre de Mersenne $M_{127} = 2^{127} - 1$.

Ronald Rivest Professeur au "Electrical Engineering and Computer Science in MIT", co-inventeur du cryptosystème RSA. Il travaille en algorithmique, intelligence artificielle et circuit VLSI.

`http ://theory.lcs.mit.edu/~rivest/`

Donald Knuth Professeur de l'université de Stanford, Knuth est l'inventeur du langage TEX un standard de l'édition scientifique. D'une manière générale, les ouvrages de Knuth sont à conseiller pour la rigueur et la profondeur du texte tant sur le plan scientifique que sur le plan historique. La présente note s'inspire par moments des commentaires des volumes I, II et III de son encyclopédie *the art of computer programming*.

`http ://www-cs-staff.stanford.edu/~knuth/`

Ron Shamir Professeur au "departement of computer sciences of the Tel Aviv university", co-auteur du système cryptographique RSA, il s'intéresse aussi à l'algorithmique génétique, la théorie des graphes et l'optimisation.

`http ://www.math.tau.ac.il/~shamir/`

Leonard Adleman Professeur à "university of south california", troisième co-auteur de RSA. Il s'intéresse maintenant à la complexité, la théorie des nombres, les mathématiques, aux calculs quantiques et autres calculs moléculaires (sic).

`http ://www-scf.usc.edu/~pwkr`

2. Computer science

Permettez moi de commencer par une petite mise au point. Nous sommes dans un cours d'informatique, et inévitablement, le titre de ce cours paraît aux yeux de certains légèrement déplacé. En effet, pourquoi parler de nombres dans un cours d'informatique ? Pythagore vous répondrait que les nombres sont la source et le principe de toute chose, alors... Pour ma part, je ne pousserai pas le bouchon si loin car ce serait prendre le risque de vous entendre dire un jour : *un ordinateur, c'est-à-dire une machine à calculer et manipuler les nombres, est capable de résoudre tous les problèmes.* Faux et archi-faux ! Qu'est-ce qu'une machine ? Que peut faire une machine ? Voilà deux interrogations fondamentales qui apparaissent en filigrane dans la liste des 23 problèmes proposés par Hilbert au début du siècle. Les réflexions de Gödel sur ces questions font émerger le caractère *indécidable* des mathématiques et accentue la « crise des fondements ». Pour arriver à ses fins, le logicien utilise les propriétés ensemblistes des nombres naturels et un argument diagonal.

Exercice 1 (argument diagonal)**.** *Par définition, un ensemble est dénombrable s'il peut être mis en bijection avec l'ensemble des entiers naturels. Montrez que l'intervalle réel* $[0, 1]$ *n'est pas dénombrable.*

Deux ingrédients qui seront réutilisés par Turing et von Neumann pour démontrer l'impossibilité de résoudre certains problèmes avec une machine mécanique. La logique et les nombres montrent que les machines à calculer pleines de circuits logiques ne peuvent pas résoudre tous les problèmes, un résultat et une preuve qui vont bien dans le sens de Pythagore...

L'argument diagonal montre que les deux infinis $\mathbf{N}$ et $\mathbf{R}$ sont non comparables. Un des mes anciens professeur, le logicien Fraïssé, nous expliquait que le plus petit ressemble à un puits infiniment profond, et le second à un océan infiniment large. L'hypothèse du continu affirme qu'il n'existe pas de cardinalité comprise entre ces deux infinis. Une assertion

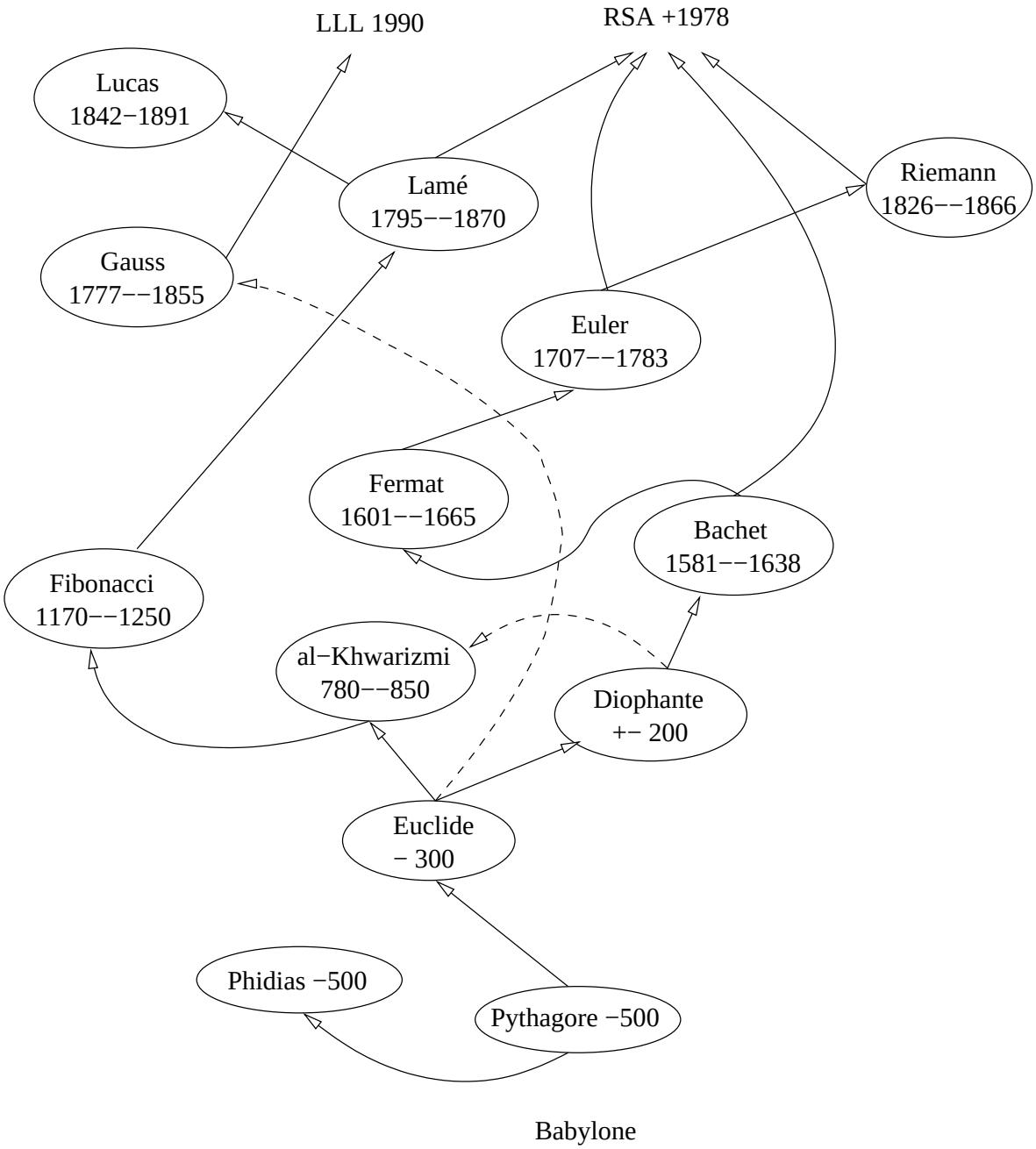


FIG. 1. RSA : généalogie.

formulée en 1878 par Georges Kantor qui consacra une grande partie de son existence à la recherche d'une preuve sans jamais y parvenir, et pour cause :

Théorème 1 (Cohen, 1963)**.** *L'hypothèse du continu est une proposition indécidable.*

Nous aurons l'occasion de constater autrement la pensée de Pythagore au travers de cette *petite histoire des nombres* qui, soit dit en passant, aurait tout aussi bien pu s'appeler RSA *: généalogie d'un cryptosystème*. Est-ce de l'informatique ? Si non alors disons que c'est de la *computer science*, de la *science des ordinateurs*.

3. Théorie et pratique

Dans notre pays, le monde [1] de la recherche et de l'enseignement n'a pas jugé bon d'introduire la terminologie strictement équivalente sciences des ordinateurs. Le mot « informatique » assemblage des mots français « information » et « automatique », ajouté au chauvinisme de mes compatriotes explique sans doute cet particularité européenne. Quoi qu'il en soit, beaucoup d'informaticiens pensent qu'il faudrait distinguer entre l'informatique pratique et une sciences des ordinateurs trop théorique pour mériter d'être enseignée aux informaticiens... Fidèle à la mémoire du précurseur Alan Turing, je ne partage pas cette idée !

I want be a computer scientist !

–Anonymous, November 1999

Le professeur Donald Knuth, de l'université de Stanford, fait partie des nombreux mathématiciens de formation qui se sont intéressés aux ordinateurs. Preuves mathématiques et programmes informatiques les conduisent à faire une certaine synthèse de la *pratique* et de la *théorie*. Dans sa conférence *Theory and Pratice*—11th World Computer Congress, 1989—Donald Knuth nous rappelle la signification des mots *théorie* et *pratique* qui viennent des mots grecs $\theta\epsilon\omega\rho\iota\alpha$ et $\pi\rho\alpha\kappa\tau\iota\kappa\eta$. Le premier (théorie) signifie « voir », le second (pratique) c'est « faire ». Voir pour mieux faire, et faire pour mieux voir, telle devrait être la devise du *computer scientist* et du scientifique en général.

Par ailleurs, cette note doit beaucoup à D. Knuth. Sur le fond, certains passages s'inspirent directement ou indirectement des premiers tomes de son extraordinaire encyclopédie *The Art of Computer Programming*, une oeuvre en cours d'élaboration. Sur la forme, le présent document est rédigé en LaTeX, une sur-couche du langage TeX le standard de publication scientifique des *computer scientists* alimenté par le générateur de fontes METAFONT. Précisons que ces logiciels développés sous la direction de Knuth sont libres, gratuits et maintenus à jour depuis une vingtaine d'année ! Les numéros de versions des programmes TeX sont attribués de sorte à converger vers π. Le premier numéro de version de TeX fût 3., le second 3.1, le troisième 3.14 etc...À chaque fois qu'un bug est découvert, il est corrigé et donne lieu à une nouvelle version de TeX dont le numéro prend une décimale supplémentaire. La commande TeX que j'ai utilisée pour compiler ces lignes affiche :

```
This is TeX, Version 3.14159 (C version 6.1)
```

c'est rassurant :-) !

4. Le réseau Internet

Le lecteur peut juger par lui même, la qualité typographique d'un document `word` ne supporte pas la comparaison avec celle d'un document TeX. Le premier est un interprète, vous voyez le texte au fur et à mesure que vous frappez des touches du clavier. Le second est un compilateur, pour écrire un texte, il faut écrire un programme, une approche qui convient (normalement) davantage aux informaticiens. L'exécution d'un programme TeX à pour résultat la mise en forme du texte dans le cadre d'une mise en page optimale. Deux approches complémentaires, et bien entendu, il vaut mieux disposer de ces deux types de logiciels sur un ordinateur. Vous avez un accès au réseau internet ? Oui, parfait ! Lancez votre navigateur favori [2]ntre nous soit dit, j'espère que votre préférence ne penche pas du côté de l'`explorer` une marque déposée, offert gratuitement pour l'achat obligatoire de `window.xyz` lors de l'acquisition d'un PC chez carrefour. Ah, j'oubliais, `window.xyz` qui est une autre marque déposée par la société microsoft. et partez dans la toile, à la recherche

[1] Un *tout petit monde* particulièrement bien décrit par le romancier et universitaire David Lodge.

[2] E

de ces logiciels et leurs documentations via les forums de discussions. Je vous conseille la suite bureautique `staroffice` et la distribution de TEX GUTenberg. L' installation de ces logiciels ne posent pas de difficultés majeures, à moins de travailler sur une `sparc` sous `linux`, sans `solaris`! Pour tester votre installation de TEX, éditez un fichier `foo.tex` [3] incluant :

```
\magnification 1200
\sl
La notation $\Theta$ introduite dans le cours d'algorithmique permet
de simplifier les formules sommatoires. En effet~:
$\sum_{i=0}^n i^k = \Theta( n^{k+1} ).$
Pour \'ecrire en \TeX, il suffit {\bf d'\'ecrire} ce qu'on {\bf lit}
sur son papier~!
\end
```

la commande `tex foo ; xdvi foo` devrait faire apparaître quelque chose comme :

La notation Θ *introduite dans le cours d'algorithmique permet de simplifier les formules sommatoires. En effet :* $\sum_{i=0}^{n} i^k = \Theta(n^{k+1})$. *Pour écrire en TEX, il suffit* **d'écrire** *ce qu'on* **lit** *sur son papier!*

Et oui, TEX est un éditeur qui permet d'écrire ce qu'on voit et lit sur son papier : une idée simple et puissante qu'on retrouve notamment dans les langages à balises tel que HTML. Cette approche s'oppose à l'édition de texte *wysiswyg* : what you see is what you get. À ce stade, le débutant reste immanquablement dubitatif... C'est clair, TEX exige de l'utilisateur un minimum de capacité d'abstraction! À vous de voir, pratiquez et forgez vous votre propre opinion.

Le réseau Internet permet le transfert d'informations à l'échelle mondiale. Plusieurs protocoles sont disponibles : ping, finger, mail, ftp, gopher, http etc... Considérons le mail par exemple, un outil sympathique qui permet aux personnes réelles ou virtuelles de s'échanger de l'information. Il est 10 :00 et Alice ne peut se rendre chez Bob à l'heure convenue, inutile de décrocher le téléphone puisque Bob lit son courrier électronique régulièrement :

```
mail Bob@euphoria.univ-tln.fr
Bob,
Comme tu le pressentais, la derniere version du syteme VII que nous avons
installee un peu rapidement sur la grappe du campus n'est pas stable.
Disons carrement buggee~!!! Je reste a la fac pour essayer d'obtenir des
infos par la hot-line.
@12c4, Alice.
```

Le message va circuler sur le réseau de l'université, pour arriver dans la mailbox de Bob. Pendant son transfert sur le réseau, le message est accessible, et peut être intercepté par n'importe qui. Eh oui, une personne mal intentionnée peut se donner les moyens de lire tous les messages qui passent devant sa machine, évidemment c'est interdit ! mais pas plus que de se garer devant l'entrée du bâtiment U. Alors, par curiosité, violons cette interdiction et observons :

@@@PASCAL, VOILA LA DERNIÈRE VERSION DE MON TEXTE SUR RSA. JE N'AI PAS SU RÉSISTER A LA TENTATION DE DÉLIRER SUR PLUSIEURS SUJETS : NOMBRES, HISTOIRE DES SCIENCES, TEX ETC... JE TE RASSURE, IL Y A DU COURS ET DES EXERCICES ! @@@MONSIEUR LE DIRECTEUR, CE MESSAGE POUR VOUS ALERTER SUR L'IMPÉRIEUSE NÉCESSITE DE RECRUTER DANS LES PLUS BREFS DÉLAIS UN INGÉNIEUR RÉSEAU SANS QUOI LES TRAVAUX-PRATIQUES D'INFORMATIQUE NE POURRONT PAS COMMENCER A LA

[3] Comprendre `toto.tex`.

DATE PREVUE.@@@JULES VIENT DE M'APPELER. IL FAIT UN BRIDGE CE SOIR, ES TU LIBRE ?@@@USERNAME WOLFMANN PASSWORD 1+1=0/F2@@@JE SORS DU CONSEIL, J'AI BIEN L'IMPRESSION QU'ON C'EST FAIT, UNE FOIS DE PLUS, ROULER DANS LA FARINE PAR NOS COLLEGUES DE LA SECTION XXVIII@@@BIZARRE, FIGURE TOI QUE MARIE-CHANTALE A PREVU UN BRIDGE CHEZ SES COPINES. OK, A CE SOIR. PS JE PREND DES PIZZAS@@@MERCI DE JETER UN OEIL SUR LE FICHIER LATEX ATTACHE, J'AIMERAIS AVOIR TON AVIS AVANT DE SOUMETTRE UN ARTICLE@@@LE REDEPLOIEMENT DES POSTES, C'EST POUR QUAND ?

Stop ! C'est une évidence, il faut se protéger. Mais comment ? Les plus conservateurs dirons qu'il vaut mieux utiliser le courrier papier, et les boîtes aux lettres en fer, c'est sûr, et en plus, tellement pratique. Ouais, d'accord, mais à condition d'acheminer son courrier soit même : les contraintes de la réalité virtuelle sont de même nature que celle du monde matériel !

5. Protection de l'information

Au XII-ième siècle, Léonard de Pise ramène de ses voyages au Moyen-Orient, le *sifr* arabe qui devient *cifra* en italien. Dans notre langue, le chiffre perd son sens *premier* « zéro » pour signifier « coder, crypter ».

Le codage et le cryptage ont pour objectif la protection de l'information lors des transmissions dans l'espace (internet, sondes spatiales) et dans le temps (disques compacts), mais à des fins très différentes. Les codes correcteurs protègent des erreurs aléatoires inhérentes au canal de transmission. La théorie des codes correcteurs est une discipline récente, elle se développe dans les années quarante dans le sillage de la théorie de l'information. Les Shannon, Golay et Hamming en sont des illustres précurseurs.

Le principe de transmission par télétype. Le jeu de caractères d'un télétype est réduit à 128 caractères. Chaque lettre est représenter par un nombre de 7 bits, c'est le fameux code ASCII. Par exemple, l'espace est représenté par 32, la lettre A par 65 etc... Soit $b_1 b_2 \dots b_7$ la lettre à transmettre. On ajoute un *bit de parité* b_8 égal à la parité du nombre de 1. Le symbole $b_1 b_2 \dots b_7 b_8$ est transmis. Lors de son acheminement des erreurs peuvent se produire, et le symbole reçu $r_1 r_2 \dots r_7 r_8$ diffère peut-être de l'octet transmis. En calculant la parité du symbole reçu, on peut avoir la certitude qu'une erreur s'est produite et dans le meilleur des cas, on peut simplement dire que le symbole est probablement correct, mais c'est déjà ça...

Exercice 2. *Décrire un protocole de transmission teletype fondé sur la détection d'erreur.*

Exercice 3 (code correcteur)**.** *Observez les* 8 *mots binaires de longueurs* 7 *:*

0000000	1001101	0101011	0010111
1100110	1011010	0111100	1110001

Leur ensemble possède une propriété remarquable qui peut être utilisée pour protéger et corriger la transmission de 8 *symboles.*

La cryptographie est vieille comme le monde, ou du moins comme l'écriture et la politique (j'imagine). Son objectif est de protéger des messages contre les intelligences malveillantes. Le plus simple des cryptosystème consiste à permuter les codes des lettres. Mais la cryptanalyse de cette méthode est facile. Quelques éléments d'information contextuelle sur le cryptogramme suffisent pour retrouver le message en clair.

Exercice 4. *Utilisez la table (*TAB.*1) des fréquences d'apparitions des différentes lettres de l'alphabet dans un texte de langue française pour « casser » le message chiffré un peu plus loin.*

TAB. 1. Fréquences des lettres dans un texte en français calculées à partir de morceaux choisis dans les œuvres de Gustave Flaubert et Jules Vernes.

A	8.40	B	1.06	C	3.03	D	4.18
E	17.26	F	1.12	G	1.27	H	0.92
I	7.34	J	0.31	K	0.05	L	6.01
M	2.96	N	7.13	O	5.26	P	3.01
Q	0.99	R	6.55	S	8.08	T	7.07
U	5.74	V	1.32	W	0.04	X	0.45
Y	0.30	Z	0.12				

Pour renforcer la méthode de chiffrement par permutation, on peut assembler les lettres pour former des digraphes, puis faire une permutation des digraphes. Face à la puissance des calculateurs, c'est encore très insuffisant. Pour faire un système de haute sécurité, il suffit de rassembler les lettres par paquets assez long, ce qui finit par poser un problème d'effectivité : comment représenter une permutation des quadrigraphes du code ASCII ?

Exercice 5. *Fabriquez des permutations de l'ensemble des quadrigraphes du code* ASCII... *Précisez la permutation inverse !*

J'ai utilisé le code TEX ci-joint pour crypter le dernier paragraphe de cette section. Utilisez les fréquences d'apparitions des lettres d'un texte français pour percer le mystère du texte qui termine cette section !

VB LCVB LJ LYYLBYJ IXYAOXNYHMOL XUALAL HQXBALLJ HZL ALGAL IL NLXYNLJ JHUI MXBY MLYABYKLY CB AYHSHFC IB ZYPMA HUHCPJLBY. QL JBFJ AYLJ LRBL IL SXBJ IFYL VBL QHF KFLU ZXRMYFJ CHBAYL JXFY VBL SXBJ HSFLE AXBQXBYJ BUL LUSFL WXCCL IL RL WHFYL IHUJLY. QL NHYIL CL JXBSLUFY IL SXAYL KHFJLY LA QL SXBIYHFJ KFLU VBL ZL JXFA CH BUL MYLBSL VBL QL MBFJJL LAYL HFRLL MHY SXBJ. QL JBFJ MYLAL H RXUAYLY RXU HWWLZAFXU AXBAL ILJFUALYLJJLL LA JHUJ ZHC' ZBC, LA JF SXBJ SXBCLE RL SXFY HBJJF SXBJ ILSXFCLY JHUJ HYAFWFZL RXU HRL AXBAL UBL, SLULE RL WHFYL BUL SFJFAL. UXBJ ZHBJLYXUJ LUAYL HRFJ, WYHUZOLRLUA. QL SXBJ MYXBSLYHF VBL QL JBJ CH WLRRL JFUZLYL, ZHMHKCL IL SXBJ XWWYFY C'HWWLZAFXU CH MCBJ MYXWXUIL ZXRRL CH MCBJ LAYXFAL LU HRFAFL, LU BU RXA CH RLFCCLBYL MYLBSL VBL SXBJ MBFJJFLE YLSLY, MBFJVBL SXAYL HRL LJA CFKYL. MLUJLE VBL CH JXCFABIL XB QOH' KFAL LJA KFLU CXUNBL, KFLU IBYL LA JXBSLUA IFWWFZFCL. HFUJF, LU P JXUNLHUA Q'HF CHRL NYXJJL. HZZXBYLE IXUZ SFAL LA SLULE RL CH WHFYL XBKCFLY MHY CHRXBY XB QL SLBG RL RLAAYL.

NLXYNLJ JHUI H HCWYLI IL RBJJLA.

```
\def\CrypteLettre#1{\carin=`#1
\carout=\carin%
\ifnum\carin>64
   \ifnum\carin<91\CrypteASCII\fi
\fi
\char\carout}
```

```
% % % CRYPTOZONE % % %
\newcount\carout \newcount\aux
\newcount\carin  \newcount\clefA
\newcount\clefB \newcount\clefR

\def\Clefs#1#2#3{\clefA=#1
\clefR=#2  \clefB=#3}

\def\CrypteASCII#1{
           \advance\carin by-65
           \carout=1
           \aux=\clefR
           \loop
       \multiply\carout by\carin
              \advance\aux by-1
            \ifnum\aux>0
            \repeat
            \multiply \carout by\clefA
            \advance\carout by\clefB
   \aux=\carout
           \divide\aux by 26
           \multiply\aux by 26
           \advance\carout by -\aux
           \advance\carout by97}
 \def\chiffre#1{{\go#1\end}}

\def\go#1{\ifx#1\end \let\next=\relax
\else\CrypteLettre{#1}\let\next=\go\fi
\next}
```

6. Cryptosystème à clefs publiques

Les premiers codes secrets sont conçus en pensant que les méthodes de chiffrement et de déchiffrement doivent être pratiquement équivalentes. La connaissance de la méthode de chiffrement implique d'une façon ou d'une autre, la connaissance de la méthode de déchiffrement. Le secret repose sur une clef robuste qui permet l'échange entre quelques services mis dans la confidence.

Dans les années 30, le mathématicien Alan Turing décrit une machine théorique (machine de Turing) pour étudier les problèmes de calculabilité. De ses travaux théoriques naît l'ordinateur. Le premier exemplaire est construit par les services du chiffre britannique pour décrypter les messages codés de la flotte nazi.

Au milieu des années soixante-dix, W. Deffie et M. Hellman proposent un nouveau concept : la cryptographie à clefs publiques, qui permet de répondre aux exigences des communications modernes. Sur internet, il faut que n'importe qui puisse communiquer en toute sécurité avec qui lui chante sans passer par un tiers quelconque. Dans ces nouveaux systèmes, la connaissance de la méthode de chiffrement ne suffit pas pour décrypter, du moins sans une consommation de temps et d'énergie exorbitante. Un tel système est fondé sur l'utilisation d'une fonction f de l'ensemble des textes clairs vers les cryptogrammes telle que $f(x)$ soit facile à calculer alors que $f^{-1}(y)$ soit pratiquement incalculable. On dit que f est une fonction à sens unique.

Exercice 6. *Donnez des exemples de fonctions qui ne sont pas à sens unique. Justifiez vos réponses. Donnez des exemples de fonctions susceptibles d'être à sens unique. Pas de preuves, juste un peu de feeling !*

Un cryptosystème à clefs publiques repose sur l'emploi d'un algorithme de chiffrement unique mais paramétrable par un ensemble de clefs. Chaque utilisateur se fabrique une clef secrète s et une clef publique k. Quand Alice souhaite envoyer un message x à Bob, elle utilise p_B, la clef publique de son collègue. Ce n'est pas difficile, tout le monde connaît la clef publique de Bob. Par contre seul Bob connaît s_B sa clef secrète. Alice transmet le cryptogramme $y := f(x, p_B)$. Si la fonction est $x \mapsto f(x, p_B)$ est à sens unique alors personne ne peut retrouver le clair x à partir de y sauf Bob qui possède le joker s_B qu'il utilise pour retrouver x via un calcul $g(y, s_B)$. Les fonctions f et g doivent satisfaire :

$$\forall k, s, x \quad g(f(x, k), s) = x$$

La sécurité du système est assurée par la réelle difficulté (admise et reconnue par la communauté scientifique) de déterminer une clef secrète connaissant une clef publique.

7. Rivest-Shamir-Adleman

Le système RSA créé au milieu des années soixante-dix par R. Rivest, R. Shamir et L. Adleman intègre la suggestion de Deffie et Hellmann. Il est fondé sur la difficulté admise de factoriser un grand nombre entier.

Fait 1. *Soient p et q deux nombres premiers de 100 chiffres décimaux, formons le produit $n = pq$, c'est un nombre de 200 chiffres. On estime qu'il faut plusieurs années de calculs pour retrouver p et q à partir de n.*

```
RSA press: http://www.rsa.com/pressbox/html/990826.html
more info: http://www.rsa.com/rsalabs/html/rsa155.html
Factorization of a 512-bits RSA key using the Number Field Sieve
-----------------------------------------------------------------
On August 22, 1999, we found that the 512-bits number

RSA-155 =
109417386415705274218097073220403576120037329454492059909138421314763 4\
```

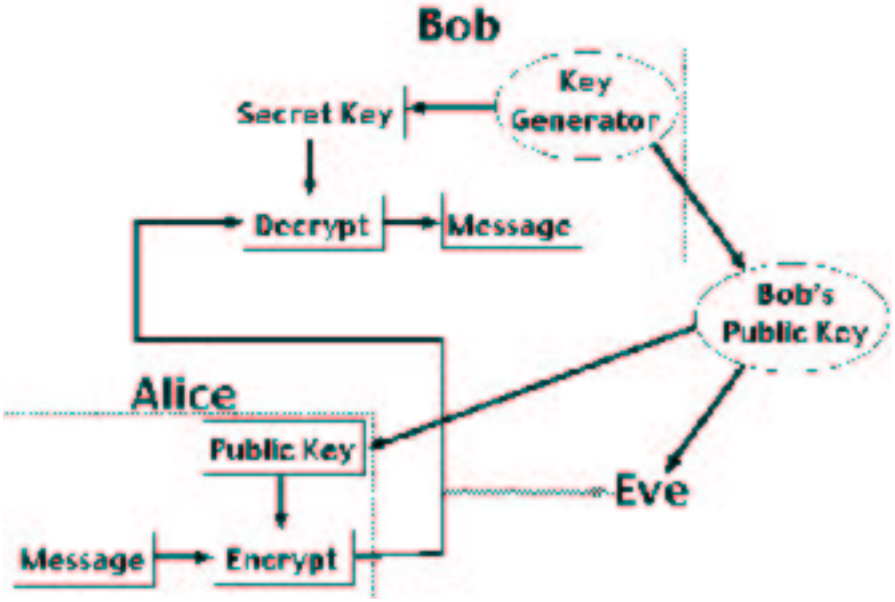


FIG. 2. Principe de la cryptographie à clef publique. *Source de l'image :* `http ://pajhome.org.uk/crypt/rsa/intro.html`

```
99842889347847179972578912673324976257528997818337970765372440271467 43\
531593354333897

can be written as the product of two 78-digit primes:

10263959282974110577205419657399167590071656780803806680334193352179071 1307779
*
10660348838016845482092722036001287867920795857598929152227060823719306 2808643

Primality of the factors was proved with the help of two different
primality proving codes. An Appendix gives the prime decompositions of p +- 1.
The number RSA-155 is taken from the RSA Challenge list
                        (see http://www.rsa.com/rsalabs/html/factoring.html).

This factorization was found using the Number Field Sieve (NFS) factoring
algorithm, and beats the 140-digit record RSA-140 that was set on February 2,
1999, also with the help of NFS [RSA140]. The amount of computer time spent
on this new factoring world record is estimated to be equivalent to 8000 mips
years. For the old 140-digit NFS-record, this effort was estimated to be
2000 mips years. Extrapolation using the asymptotic complexity formula for NFS
would predict approximately 14000 mips years for RSA-155. The gain is caused
by an improved application of the polynomial search method used for RSA-140.
```

Exercice 7. *La méthode du crible quadratique de Fermat, améliorée par Pomerance au début des années* 80 *débouche sur un algorithme de complexité*

$$O(e^{\sqrt{\log n \log\log n}}),$$

pour factoriser un entier n*. Le* 22 *Aout* 1999*, un entier* RSA *de* 155 *chiffres décimaux a été factorisé en* 4 *mois par* 300 *machines ce qui équivaut à un travail de* 8000 *MIPS-année. Estimez le temps de calcul de factorisation d'un entier* RSA *de* 200 *chiffres.*

L'utilisateur de RSA commence par fabriquer deux nombres premiers p et q de grandes tailles : une centaine de chiffres décimaux. Il peut y arriver :

Fait 2. *Il existe beaucoup de nombre premier.*

Démonstration. Dans la section (SEC.11), nous verrons que la proportion de nombres premiers de moins de 100 chiffres décimaux est de l'ordre de $\frac{1}{250}$. □

Fait 3. *On sait tester très rapidement la (pseudo)-primalité d'un nombre.*

Démonstration. Il s'agit d'une généralisation du petit théorème de Fermat, (SEC.9). □

Il calcule le produit $n = pq$. Il existe $\phi(n) = (p-1)(q-1) = n - (p+q) + 1$ nombres entiers inférieurs à n qui sont premiers avec n. Un petit théorème d'Euler affirme que :

Théorème 2 (Euler)**.** *Soit x un entier. Si x est premier avec n alors $x^{\phi(n)} \equiv 1 \mod n$.*

L'entier n constitue une part de la clef publique, sans révéler la valeur de p et de q. Ensuite, Bob choisit un entier k au hasard mais premier avec $\phi(n)$. Cet entier possède un inverse s modulo $\phi(n)$ qui vérifie

$$ks \equiv 1 \mod \phi(n)$$

L'entier k constitue la seconde partie de la clef publique, l'entier s est la clef secrète. Il faut remarquer l'effectivité de ce point, l'algorithme dit du « pgcd étendu » permet de faire le calcul de k en $\Theta(\log n)$ divisions euclidiennes dont les opérandes ne dépassent pas les 200 chiffres.

Le clair x est chiffré par le calcul $f(x,k,n) = x^k \mod n$ et le cryptogramme y est décrypté de la même manière par le calcul $g(y,s,n) = y^s \mod n$.

Fait 4. *Les procédures de cryptage et décryptage sont effectives puisque le calcul de x^t modulo n nécessite $\Theta(\log(n))$ multiplication et réductions modulaires.*

Fait 5. *Calculer s à partir de k et n revient à factoriser l'entier n. La sécurité du système est assurée par le fait numéro 1.*

algorithme	action	algorithme	action
`ZERO(x : REGISTRE)`	$x := 0$	`NUL(x : REGISTRE)`	$x = 0\,?$
`UN(x : REGISTRE)`	$x := 1$	`ONE(x : REGISTRE)`	$x = 1\,?$
`COPIER(x,y : REG.)`	$x := y$	`CMP(x, y : REG.)`	$x = y\,?$

TAB. 2. Opérations de bases

8. Registres de calculs

```
FONCTION CMP( x, y : REGISTRE)
//  retourne -1, 0 ou +1 suivant
// la valeur de x est < = ou > a
// celle de y.
VARIABLE i : INDICE;
DEBUT
i :=  N;
TANTQUE (x[i] = y[i]) ET (i>=0)
    FAIRE DEC(i);
FINTQ
SI ( i < 0 )
    RETOURNER( 0)
FINSI
SI (x[i]>y[i])
    RETOURNER(+1)
SINON
    RETOURNER(-1)
FINSI
FIN
```

RSA est facile à mettre en oeuvre à condition de disposer des opérations usuelles sur les nombres. Les langages de calculs formels manipulent des entiers avec une précision infinie (i.e. arbitraire) et facilite l'implantation de l'expon–entiation modulaire. Les registres des mini-ordinateurs sont sur 64 bits, sans effort, nous pouvons nous amuser à faire du RSA avec des nombres de 64 chiffres binaires, c'est-à-dire une dizaine de chiffres décimaux. Mais là, nous sommes bien loin des 200 chiffres préconisés ! Pour aller plus loin, nous devons implanter des registres plus grands.

Fixons une bonne fois pour toute la valeur d'une base B, du point de vue structure de données, un entier RSA peut-être considéré comme un tableau de chiffres. Nous dirons que l'entier M est de taille N lorsqu'il est représentable par un tableau de taille N. Bien entendu, la taille de M dépend du logarithme :

$$B^{N-1} \leq M < B^N \Longrightarrow \lfloor \log_B M \rfloor = N - 1.$$

On définit le type `REGISTRE = TABLEAU[N] CHIFFRE`, et nous savons depuis l'école primaire qu'il est possible de calculer les 4 opérations élémentaires par des mécanismes de « retenues ». De ces quatre, opérations nous en déduisons les procédures et fonctions nécessaires à la mise en oeuvre de RSA.

Pour nous familiariser avec cette représentation, écrivons par exemple la fonction de comparaison `CMP(x,y)` qui teste si le contenu du registre x est supérieur ou égal à celui de y.

L'algorithme ne doit pas poser de problème au lecteur qui en comprend parfaitement l'aspect syntaxique et logique. En est-il de même du point de vue du temps de calcul ? La complexité du cas favorable est constante, celle du pire des cas est linéaire, et en moyenne l'utilisation de `CMP(x,y)` est-elle : plutôt constante ou plutôt linéaire ?

```
PRD-MOD(x,y,z: REGISTRE)
VARIABLE i : INDICE;
         aux: REGISTRE;
DEBUT
ZERO(aux);
i := 0;
TANTQUE (i< N) FAIRE
   XADD( aux, y[i], x);
   REDUCTION(aux, z)
   B-MUL(x)
   REDUCTION(x, z);
   INC(i);
FINTQ
COPIER( x , aux);
FIN
```

Exercice 8. *Précisez les instances favorables et défavorables de l'algorithme `CMP(x,y)`. Calculez la complexité moyenne.*

Il est impensable d'écrire un programme sans initialisation, test et affectation ! C'est le rôle des opérations basiques ci-dessus (TAB.2) que d'initialiser, tester et affecter les variables de type `registre`. La présence des éléments « neutres » dans ces définitions est loin d'être fortuite ! La fonction `NUL(x)` compare x à 0 alors que la procédures `ZERO(x)` initialise x à 0 et que `COPIER(x,y)` affecte la valeur de y à x.

Exercice 9. *Quels sont les temps de calcul des opérations basiques ?*

`INC(x :REGISTRE)`	$x := x + 1$	$\mathrm{O}(N)$
`ADD(x,y :REGISTRE)`	$x := x + y$	$\Theta(N)$
`STS(x,y :REGISTRE)`	$x := x - y$	$\Theta(N)$
`C-MUL(x :REGISTRE, c :CHIFFRE)`	$x := cx$	$\Theta(N)$
`C-DIV(x :REGISTRE, c :CHIFFRE)`	$x := x/c$	$\Theta(N)$
`DICHO(x :REGISTRE)`	$x := x/2$	$\Theta(N)$
`B-MUL(x :REGISTRE)`	$x := Bx$	$\Theta(N)$
`B-DIV(x :REGISTRE)`	$x := x/B$	$\Theta(N)$
`XADD(x,y :REGISTRE, c :CHIFFRE)`	$x := x + cy$	$\Theta(N)$
`XSTS(x,y :REGISTRE, c :CHIFFRE)`	$x := x - cy$	$\Theta(N)$
`PRD(x,y :REGISTRE)`	$x := xy$	$\Theta(N^2)$

TAB. 3. Opérations élémentaires

L'algorithme de multiplication modulaire `PRD-MOD(x,y,z)` effectue l'opération $x \leftarrow (x * y) \mod z$, opération cruciale pour RSA. Il est implanté à l'aide des primitives décrites dans la table (TAB.3) qui nous renseigne aussi sur les temps de calcul. À partir de ces opérations, nous définissons les opérations modulaires. Nous verrons plus loin que la complexité de l'algorithme de réduction est quadratique, d'où celle des autres opérations (TAB.4).

Exercice 10. *Précisez les instances favorables et défavorables de l'algorithme* ***`INC(x)`****. Calculez la complexité moyenne.*

Exercice 11. *L'algorithme* ***`DICHO(x)`*** *divise x par 2 : idéal pour jouer à « quel est l'âge du capitaine ». Donnez en une version lorsque la base B est paire pour en déduire un algorithme de calcul de racine carrée.*

Exercice 12. *Proposez plusieurs algorithmes d'extraction de racine carrée, comparez les du point de vue temps de calcul.*

Exercice 13. *Implantez ces algorithmes !*

algorithme	action	complexité
`REDUCTION(x, z :REGISTRE)`	$x := x \mod z$	$\Theta(N^2)$
`PRD-MOD(x,y,z :REGISTRE)`	$x := x * y \mod z$	$\Theta(N^2)$
`CARRE-MOD(x,z :REGISTRE)`	$x := x * x \mod z$	$\Theta(N^2)$
`EXP-MOD(x, z :REGISTRE, c :CHIFFRE)`	$x := x^c \mod z$	$\mathrm{O}(N^2 \log(B))$
`GRAND-EXP-MOD(x,y,z :REGISTRE)`	$x := x^y \mod z$	$\mathrm{O}(N^3 \log(B))$

TAB. 4. Opérations modulaires

9. Grand et petit théorèmes de Fermat

La théorie des nombres n'est pas de reste. Les progrès technologiques sont considérables à l'image de la « pascaline » de Pascal mais il faudra encore attendre une centaine d'années pour voir apparaître l'ancêtre de l'ordinateur : le métier à tisser D'Émile Jacquard.

[...] J'appelai cette manière de démontrer la *descente infinie* ou *indéfinie*; je m'en servis au commencement que pour démontrer des propositions négatives[...] J'ai ensuite considéré certaines questions qui, bien que négatives, ne restent de recevoir très grande difficulté, la méthode pour y pratiquer la *descente* étant tout à fait diverse des précédentes, comme il sera aisé d'éprouver. Telles sont les suivantes :
(1) Il n'y a aucun cube divisible en deux cubes.
(4) Toutes les puissances carrées de 2, augmentées de l'unité, sont nombres premier.
Voilà sommairement le compte de mes rêveries sur le sujet des nombres. Je l'ai écrit que parce que j'appréhende que le loisir d'étendre et de mettre au long toutes ces démonstrations et ces méthodes me manquera; en tout cas, cette indication servira aux savants pour trouver eux-mêmes ce que je n'étends point...
Multi pertransibunt et augebitur scientia.

La contribution arithmétique au siècle des lumières est assurée par le Toulousain Pierre de Fermat dont le nom reste encore aujourd'hui très populaire, en partie à cause de la fameuse conjecture de Fermat, dite « grand théorème de Fermat » démontrée en réalité par A. Wiles, en 1995.

Théorème Soit n un entier supérieur ou égal à 3. Soient x, y et z trois entiers. Si $x^n + y^n = z^n$ alors l'un des trois nombres x, y ou z est nul.

Au début du XVII-ième siècle, Pierre de Fermat énonce cette proposition[4] dans la marge de son exemplaire de Diophante et ajoute ne pas avoir assez de place pour y écrire la preuve... Il aura fallu plus de trois siècles, une construction mathématique gigantesque et une « touche » finale d'Andrew Wiles [5] : deux années de travail acharné, et plusieurs centaines de pages de démonstration pour enfin tordre le coup à cette conjecture ! Personne ne doute de la bonne foi de Fermat lorsqu'il écrit cette note, mais la preuve qu'il avait à l'esprit fondée sur son principe de descente infinie était certainement fausse c'est du moins ce que nous pouvons penser à la lecture du court passage extrait du « testament » qu'il adresse à Carcavi en 1659. La proposition (1) n'est autre que le cas $n = 3$ du grand théorème. Dans la proposition (4) Fermat dit que les nombres de la forme $F_n = 2^{2^n} + 1$ sont tous premiers, c'est faux !

Exercice 14. *Calculez les puissances de* 2 *modulo* 641. F_5 *est-il premier ?*

Quoi qu'il en soit le Toulousain Pierre de Fermat a breveté complètement et correctement un grand nombre de résultats. Notamment, le petit théorème de Fermat[6] :

Proposition 1 (Fermat). *Soit p un nombre premier. Si a n'est pas divisible par p alors $a^{p-1} \equiv 1 \mod p$.*

Exercice 15. *What about the proof of LTF ?*

```
unsigned long int n,x,i;
n = ( 1LU ) << (32);
n = n+1; x = 3;
printf("\nt=%d",sizeof(n));
printf("\nn=%lu\n", n);
for( i = 0; i <= 32; i++)
   { printf("\n%lu", x);
   x = ( x*x) % n; }
printf("\nBye...\n", n);
```

Le programme C ci-contre affiche : $t = 8$, $n = 4294967297$ puis les 33 nombres 3,

9,	81,	6561,	43046721,
3793201458,	1461798105,	852385491,	547249794,
1194573931,	2171923848,	3995994998,	2840704206,
1980848889,	2331116839,	2121054614,	2259349256,
1861782498,	1513400831,	2897320357,	367100590,
2192730157,	2050943431,	2206192234,	2861695674,
2995335231,	3422723814,	3416557920,	3938027619,
2357699199,	1676826986,	10324303,	3029026160.

[4] LTF : Last Theorem of Fermat
[5] Je vous conseille au passage l'excellent documentaire de Simon Singh filmé par la BBC.
[6] LTF : Little Theorem of Fermat

On en déduit (sans intervention de l'oracle 641) que F_5 n'est pas premier. Est-ce correct ? Comment auriez-vous fait avec une machine 32 bits ?

10. Le théorème Chinois

Nous avons dit peu de choses des mathématiques non occidentales. À vrai dire, je n'en connais pas grand chose ! Il semble que la science chinoise très en avance sur la science occidentale ait un jour été victime de l'utilitarisme. Le théorème « chinois » ou plus précisément théorème des restes chinois apparaît pour la première fois dans le traité Juzhang Suhanshu écrit entre 280 et 473 sous la forme d'un petit problème :

Nous avons des choses dont nous ne connaissons pas le nombre ; si nous les comptons par paquets de trois, le reste est 2 *; si nous les comptons par paquets de cinq, le reste est* 3 *; si nous les comptons par paquets de sept, le reste est* 2*. Combien y a-t-il de choses ? La réponse est égal au nombre magique* 23*.*

Exercice 16. *Expliquez pourquoi le résultat est* 23*. À l'occasion, documentez vous sur le nombre* 23 *et vous comprendrez pourquoi je dis « magique ».*

La compréhension du théorème chinois est fondamentale pour qui veut pratiquer la théorie des nombres. Si c'est votre cas, je vous conseille de ne pas sortir de cette section avant d'avoir acquis des certitudes.

Pour tout entier n désignons par $\mathbf{Z}/(n)$ l'anneau des résidus modulo n. C'est l'ensemble $\{0, 1, \ldots, n-1\}$ muni des lois « quotient » d'addition $\oplus$, et multiplication $\otimes$. En notant $x \mod n$ le reste de la division Euclidienne de x par n, ces lois sont définies par :

$$x \oplus y = (x + y) \mod n, \quad \text{et} \quad x \otimes y = (x \times y) \mod n$$

Proposition 2. *Soient a et b deux nombres entiers premiers entre eux. L'application $x \mapsto (x \mod a, x \mod b)$ qui applique $\mathbf{Z}/(ab)$ dans $\mathbf{Z}/(a) \times \mathbf{Z}/(b)$ est un isomorphisme d'anneaux.*

Démonstration. Les deux ensembles ont le même cardinal, il suffit de montrer la surjectivité ou l'injectivité. L'injectivité résulte du lemme d'Euclide et la surjectivité est une affaire de cardinalité. □

La démonstration proposée ci-dessus est correcte mais non constructive. Elle ne dit pas comment construire l'image réciproque d'un couple (x, y).

Exercice 17. *Utilisez le théorème de Bachet pour déterminer l'image réciproque des couples* $(1, 0)$ *et* $(0, 1)$*, en déduire celle d'un couple* (x, y)*.*

Le théorème chinois se généralise sans peine. Soit $M = \prod_{i=1}^{k} M_i$ le produit de k entiers premiers entre-eux deux à deux. L'anneau $\mathbf{Z}/(M)$ est isomorphe à l'anneau produit :

$$\mathbf{Z}/(M_1) \times \mathbf{Z}/(M_2) \times \cdots \times \mathbf{Z}/(M_k).$$

Exercice 18. *L'isomorphisme précédent peut être utilisé pour faire des calculs modulo un grand entier composé de petits facteurs premiers. Expliquez comment et pour quelles types d'applications.*

11. La contribution d'Euler

Un petit siècle après Fermat, Euler poursuit (entre autres) les travaux de Fermat. Il trouve des erreurs dans le testament de son prédécesseur, en particulier doute de la validité de la proposition (*). Finalement, Euler prouve que le cinquième nombre $2^{2^5} + 1 = 4294967297$ n'est pas premier ! Il valide le troisième cas du grand théorème et généralise le petit théorème de Fermat ce qui est capital pour RSA.

Soit n un entier non-nul, le nombre d'entiers x compris entre 1 et n premier avec n est égal à $\varphi(n)$, où φ est une fonction multiplicative complètement déterminé par $\varphi(p^r) = (p-1)p^{r-1}$.

Proposition 3 (Euler). *Soit n un entier. Si a est premier avec n alors $a^{\varphi(n)} \equiv 1 \mod n$.*

Démonstration. C'est bien entendu une conséquence du théorème de Lagrange, un résultat que ne pouvait pas connaître Euler ! □

Exercice 19. *Démontrez le résultat précédent dans le cas qui nous intéresse i.e. quand n est multiple de deux nombres premiers. Utilisez le théorème Chinois !*

Exercice 20. *On suppose toujours que n est un entier* RSA. *Déduisez de l'exercice précédent la valeur de $x^{\varphi(n)}$ modulo n, pour tous les entiers x inférieurs à n.*

Les champs d'investigation d'Euler sont vastes. La résolution du puzzle des ponts de Königsberg qui est à l'origine de la théorie des graphes, et de l'analyse « in situ » i.e. la géométrie des formes et qui contient les germes de la topologie algébrique. Les grecs dont Platon qui vénéraient les polyèdres réguliers auraient apprécié le théorème fondamental d'Euler qui affirme que dans un polyèdre convexe la somme alternée des faces, arêtes et sommets est égal à 2. Les mathématiques de l'époque d'Euler manque de puissance, pour s'exprimer librement, Euler est obligé d'oser des raisonnements faux qui s'éclairciront au fil du temps. L'un d'entre-eux nous concerne de très prés puisqu'il conduit à la distribution des nombres premiers. Pour Euler, la méthode du crible d'Eratostène correspond à l'égalité formelle :

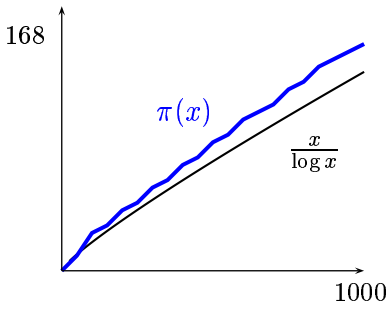
$$\sum_{0\leq n} \frac{1}{n} = \prod_{p} \frac{1}{1-\frac{1}{p}} \tag{1}$$

d'où il tire que si p est assez grand alors la somme $\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{p}$ est asymptotiquement équivalente à $\log\log p$. De sorte que si nous notons $\pi(x)$ le nombre de premiers inférieurs ou égaux à x alors

$$\pi(x) \sim \int_2^x \frac{dt}{\log t} \sim \frac{x}{\log x}$$

(théorème des nombres premiers)

La validité de cette formule est démontrée par Hadamard et De La Vallée-Poussin quelques décennies plus tard.



Le lecteur est invité à coder la méthode du crible d'Eratostène pour obtenir le graphe de la fonction $x \mapsto \pi(x)$ et par là même se convaincre des raisonnements de Gauss et Euler.

EXERCICE 1. Quelle est la complexité de l'algorithme du crible d'Eratostène ?

La théorie des séries et des fonctions analytiques permet de donner un sens aux calculs d'Euler. Pour l'essentiel, disons que la série du membre gauche de l'égalité (1) ne converge pas pour y remédier, Bernard Riemann introduit sa fameuse fonction zêta définie sur une partie adéquate du plan complexe par

$$\zeta(z) = \sum_{0\leq n} \frac{1}{n^z}$$

une fonction n'est pas sans mystère puisqu'elle conduit à l'expression d'une conjecture stupéfiante connue sous le nom d'hypothèse de Riemann. On utilise la fonction Gamma

définie par $\Gamma(s) = \int_0^\infty e^{-t} t^s \frac{dt}{t}$ pour obtenir un prolongement analytique de la fonction zêta :

$$\mathrm{Z}(s) = \int_0^\infty \frac{t^s}{e^t - 1} \frac{dt}{t}$$

qui est méromorphe sur $\mathbf{C}$ avec un unique pôle en 1 de résidu 1. Tous les nombres pairs négatifs sont des zéros de Z, et

Conjecture 1 (Riemann). *L'hypothèse de Riemann affirme que les autres zéros sont sur l'axe formé des nombres complexes de partie réelle* $\frac{1}{2}$.

Une conjecture qui peut être reformulée plus concrètement en terme de répartition des nombres premiers. En effet, l'hypothèse de Riemann est équivalente à la formule :

$$\pi(x) = \mathrm{Li}(x) + \mathrm{O}(\sqrt{x} \log x)$$

La démonstration de cette conjecture serait considérer comme une sorte d'achèvement de l'histoire des nombres. Elle est au programme des 23 problèmes de Hilbert, ce qui implique que la plupart des meilleurs mathématiciens se soient intéressés à cette question sans aboutir... Elle est au programme des sept problèmes du millénaire présentés le 24 Mai 2000 au collège de France [2] ... Avis aux amateurs !

12. Exponentiation modulaire et Pseudo-primalité

```
BIG-EXP-MOD(x,y,z: REG.)
// Calcul de x^y modulo z
VARIABLE     i : INDICE;
           u, v: REGISTRE;
DEBUT
COPIER(u,x)
EXP-MOD(x, y[0], z)
i := 1
TANTQUE ( i < N ) FAIRE
    EXP-MOD(u, B, z)
    SI ( y[i] > 0) ALORS
       COPIER( v, u)
       EXP-MOD(v, y[i],z)
       PRD-MOD(x, v, z )
     FINSI
    INC(i);
FINTQ
FIN
```

L'algorithme `BIG-EXP-MOD`(x, y, z) calcule $x^y \mod z$ pour des grands entiers x et z et un grand exposant y. Il utilise un algorithme `EXP-MOD(x, c)` qui fait la même chose mais pour un exposant inférieur ou égal à la base B. Une implantation optimale de `EXP-MOD()` produit au plus $\log_2(B)$ itérations d'où la complexité $\mathrm{O}(N^2 \log_2(B))$.

Exercice 2. Montrez que `BIG-EXP-MOD`(x, y, z) est de complexité cubique.

Exercice 3. Utilisez `DICHO(y)` et l'élévation au carré `CARRE-MOD(x, z)` pour écrire directement un algorithme `BIG-EXP-MOD(x,y,z)`

La section précédente nous dit qu'il existe une infinité de nombres premier et que la probabilité pour qu'un nombre de 100 chiffres choisi au hasard soit premier est de l'ordre de 4/1000. Malheureusement, il n'existe pas d'algorithme pour décider rapidement du caractère premier d'un nombre. L'algorithme naïf qui consiste à tester la divisibilité par les plus petits nombres est définitivement impraticable ! Nous devons rechercher notre salut dans les méthodes probabilistes. Curieusement, nous pouvons être quasiment certain de la primalité d'un nombre à condition d'avoir à l'esprit une réflexion du mathématicien Emile Borel :

Un phénomène dont la probabilité
est de 10^{-50} *ne se produira donc jamais,*
ou du moins ne sera jamais observé.

–E. Borel, Les probabilités et la vie.

Le petit théorème de Fermat suggère une procédure pour tester la primalité d'un nombre. Soit n un entier, on dit que a est un témoin de Fermat si $a^{n-1} \equiv 1 \mod n$. Le théorème de Fermat montre que si n est premier alors tous les entiers a compris entrez 1 et $n-1$ sont des témoins de Fermat. Réciproquement, on peut démontrer que 2 n'est pas un témoin de Fermat pour une majorité des nombres impairs. Malheureusement, il existe des nombres impairs qui ne possède pas assez de témoins de Fermat et qui passent au travers de ce test de pseudo-primalité.

```
TEST-DE-FERMAT(a , m )
PARAMETRE a : REGISTRE;
          m : REGISTRE;
VARIABLE  b : REGISTRE;
DEBUT
     COPIER(b,m);
     DEC(m);
     BIG-EXP-MOD(a, m, b)
RETOURNER( ONE(a) )
FIN
```

Exercice 21. *Utilisez le test de Fermat pour écrire tester la primalité des nombres $n!+1$, pour les entiers n variant de 1 à 100.*

Supposons que $n-1 = 2^r t$ avec t impair. Si $a^{n-1} = 1 \mod n$ c'est que $x^t = 1$, ou bien qu'il existe j tel que $0 \leq j \leq r$ et tel que $x^{2^j t}$ soit d'ordre 2. Lorsque n est premier cet élément d'ordre 2 est nécessairement -1.

Exercice 22. *On suppose que l'entier n est divisible par exactement r distincts nombres premiers impairs. Combien y a-t-il d'éléments d'ordre 2 dans le groupe des unités de $\mathbf{Z}/(n)$?*

On dit que a est un témoin de Rabin-Miller pour l'entier n si $x^t = 1$ ou s'il existe j tel que $x^{2^j t} = -1$. Le théorème qui suit est très efficace pour décider du caractère premier ou non-premier d'un nombre entier.

Proposition 4 (Rabin-Miller)**.** *Soit n un nombre impair. Si n n'est pas premier alors la probabilité pour qu'un entier soit un témoin de Rabin-Miller de n est inférieure à $\frac{1}{4}$.*

Démonstration. La preuve un petit peu délicate, n'est pas une priorité de ce cours. Le lecteur curieux peut consulter le petit livre de Michel Demazure □

```
RABIN-MILLER(b : REGISTRE)
VARIABLE   a,t : REGISTRE;
             r : INDICE;
            ok : BOOLEAN;
DEBUT
   COPIER(t,b); DEC(t);
   TANTQUE PAIR(t)
       DICHO(t);
       r := r + 1;
   FINTQ
   ok := VRAI; a := HASARD();
   BIG-EXP-MOD( a , t, b)
   TANTQUE ( r >= 0) FAIRE
      SI ONE(a) ALORS
         RETOURNER (ok);
      FSI
      ok := MOINS-UN(a, b);
      CARRE-MOD( a, b);
      r := r - 1;
   FINTQ
   RETOURNER(FAUX);
FIN
```

L'algorithme de `RABIN-MILLER`(b) prend en entrée un grand entier b. Il commence par déterminer la valuation dyadique de $b-1$, notée r. Il utilise une fonction booléenne `MOINS-UN`(x, b) qui retourne `vrai` si et seulement si x est égal à -1 modulo b.

EXERCICE 4. Écrivez une version de `MOINS-UN`(x, b). Un entier a est tiré au hasard, la boucle qui suit calcule $a^{2^j t}$ pour j dans $\{0, 1, \ldots, r\}$. Si un des j satisfait $a^{2^j t} = 1$ on utilise la variable `ok` pour vérifier le que $a^{2^{j-1} t} = -1 \mod b$. Si c'est le cas, un témoin de Rabin-Miller est détecté.

EXERCICE 5. La loterie nationale arnaque deux fois par semaines plusieurs millions de français. Sachant qu'un parieur mise 2Frs sur une combinaison de 6 nombres parmi 49, calculez sa probabilité de gain. Quelle est son espérance de gain ?

EXERCICE 6. Vous exécutez 25 appels successifs de

la fonction `RABIN-MILLER()`, toutes ses requêtes renvoient 1. Êtes-vous prêts à parier sur la primalité de m ?

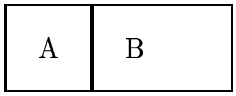
EXERCICE 7. Prouvez le théorème de Rabin-Miller lorsque n est une puissance d'un nombre premier. Essayez dans le cas d'un nombre composé congru à 3 modulo 4.

13. Le nombre d'or

Une des caractéristiques du langage TEX est la manipulation des boîtes. Le caractère A est dans une boîte que je peux matérialiser par la « macro »

```
\def\misenboite#1{\vbox{\hrule\hbox{\vrule#1\vrule}\hrule}}
```

qui appliquée à A donne A. TEX permet de spécifier des boites [7] . de différentes tailles Par exemple, je peux mettre le caractère A dans une boîte de dimension 32pt par 32pt, la lettre B dans une boîte 32pt par 52pt et mettre les deux boîtes côte à côte :



d'après Phidias, c'est une façon élégante d'agencer les deux rectangles A et B, car le ratio de la surface de leur union par l'aire du plus grand est égal au ratio de la surface du plus grand par celle du plus petit. Prenons pour unité l'aire de A, désignons par ϕ l'aire de B, pour respectez la divine proportion ϕ doit satisfaire à

$$\frac{1+\phi}{\phi} = \frac{\phi}{1}, \quad \text{i.e.} \quad \phi^2 = \phi + 1.$$

une équation du second degré qui admet deux racines réelles, l'une est positive c'est le nombre d'or $\phi = \frac{1+\sqrt{5}}{2} = 1.618...$, la seconde est négative $\hat{\phi} = \frac{1-\sqrt{5}}{2}$.

Exercice 23. *Utiliser les formules d'Euler pour mettre en équation le cosinus de* $2\pi/5$. *En déduire que le pentagone régulier à cinq cotés est constructible à la règle et au compas.*

Phénomène étrange, tous les polygones réguliers ne sont pas constructible avec une règle et un compas. Par exemple, le pentagone est constructible alors que l'heptagone ne l'est pas. Carl Friedrich Gauss a percé le mystère : pour un nombre premier impair n, le n-gone régulier est constructible à la règle et au compas si et seulement si $n-1$ est une puissance de 2, un résultat qui s'interprète joliment par la théorie de Galois, mais là c'est le début d'une autre histoire...

Exercice 24. *Que pensez vous de la suite récurrente* (x_n) *dont le terme initial est* x_1 *et où* $x_n = \sqrt{1+x_n}$ *?*

Un petit joyau pour terminer cette section. Notons $(x_n)_{n\in\mathbf{N}}$ la suite de l'intervalle réel $[0,1]$ définie par $x_n = n/\phi \mod 1$, réduire modulo 1, c'est prendre la partie fractionnaire. À l'étape n, les n éléments x_0, x_1, ... , x_{n-1} découpent l'intervalle $[0,1]$ en $n-1$ segments. On démontre, et c'est loin d'être « anisitrottant » que le réel x_n tombe à chaque fois dans le plus grand intervalle et qu'en plus il découpe ce dernier suivant divine proportion!!!

[7] Le joueur d'échecs, Aaron Nimzovitch, auteur des célèbres *Mon Système* et *La Pratique de mon Système* aurait certainement utilisé ici une *mise en parallèle à effet comique* du genre : TEX, un langage qui permet l'art de la mise en boîte... Mais là, c'est sûr, je m'égare !

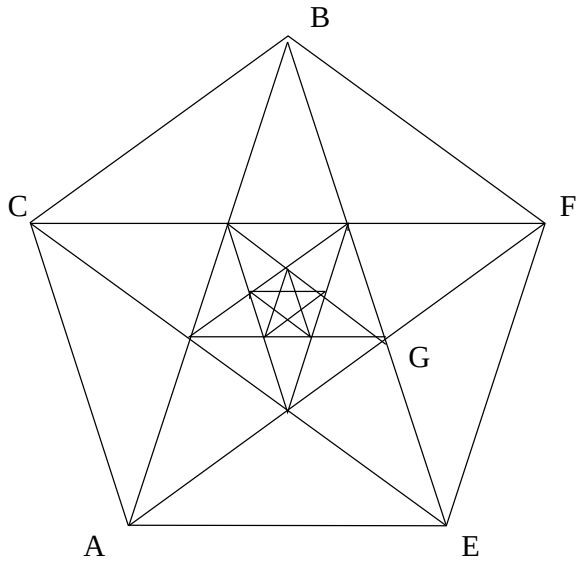


FIG. 3. Le pentogramme des Pythagoriciens

14. Le pentogramme des pythagoriciens

Pendant la période Pythagoricienne, autour de -500 av J.-C. , l'esprit des mathématiques imprègne les disciplines fondamentales : la philosophie, l'architecture. Qui de Phidias ou de Pythagore a-t-il influencé l'autre ? Le pentagone régulier est le symbole de ralliement des pythagoriciens, présent dans la nature, il est constructible à la règle et au compas, il intègre trois résultats fondamentaux de la Grèce antique. Le théorème de Pythagore permet la construction de $\sqrt{5}$ et donc celle du pentagone. Observez la figure (3), des parallèles bien choisies montrent la similitude des triangles (A, B, C) et (E, F, G). Le théorème de Thalès s'applique

$$\frac{AB}{AC} = \frac{EF}{EG} = \frac{AC}{AB - AC}$$

c'est une nouvelle fois le nombre d'or ! La suite infinie de pentagones qui se déduit de cette figure, suggère à Hépitarque une démonstration de *l'incommensurabilité* du nombre d'or : c'est un argument de « descente infinie ».

Comme l'écrit Platon, des résultats similaires concernant $\sqrt{2}$, $\sqrt{3}$ et $\sqrt{5}$ sont bien connus depuis Babylone, mais il n'existe pas de procédé clair pour faire apparaître le caractère irrationnel de tous ces nombres.

Exercice 25. *L'hypoténuse du triangle rectangle de petits côtés de mesure* 1 *et* 2 *tracé sur le sable impose la relation* $x^2 = 5$. *Partez de cette égalité pour obtenir l'irrationalité de* $\sqrt{5}$.

Mais qu'est-ce qu'un nombre ? Pour l'étudiant du XX-ième siècle, il n'y a pas lieu de réellement s'attarder sur cette question. Disons simplement qu'il y a des ensembles et des éléments, et finalement, pas vraiment de *différences* entre les notions de nombres et d'éléments. Tout est nombre en *somme* ! Quitte à *diviser* les lecteurs, je ferai remarqué que la *multiplicité* des genres : entier, relatif, rationnel, réel, complexe, p-adique, algébrique, transcendant etc... ne doit pas nous faire oublier qu'au départ, il y a une notion purement intuitive : les entiers naturels, et que tous les autres s'en déduisent logiquement à partir d'une étude systématique des propriétés des quatre opérations fondamentales : addition, soustraction, multiplication et division. Les nombres relatifs naissent de l'obstruction à soustraire, les rationnels de l'impossibilité de diviser. Les imaginaires de l'absence de carrés négatifs etc...

Exercice 26. *Soit Σ la série $1+2+4+8+\ldots$. Imaginons que cette série converge. La différence entre 2Σ et Σ vaut -1 et donc : $1+2+4+8+\cdots=-1$?Mais une ne série de termes positifs ne peut pas converger vers un nombre négatif!!! Bref, tout ça est bien absurde...À moins que... ?*

Die Zahlen sind freie Schöpfungen des menschlichen Geites,
sie dienen als ein Mittel, um die Verschiedenheit der Dinge leichter
und schärfer aufzufassen.

–Dedekind, Was Sind und was sollen die Zalhen, 1887

15. La récursivité et les entiers naturels

Nous connaissons tous l'ensemble des entiers naturels : *un*, *deux*, *trois*, *quatre*, *plusieurs*, sans *rien* oublier. Voilà notre perception primitive des nombres, un point de vue que nous partageons probablement avec d'autres espèces du genre animal. L'invention du concept de nombre permet de faire reculer les frontières. Sans les percevoir réellement, nous sommes en mesure d'appréhender de nouveaux nombres : zéro, cinq etc...

Die ganzen Zahlen hat der liebe Gott gemacht,
alles andere ist Menschenwerk.

–Leopold Kronecker, Jahresber.

L'*infini* qui se cache derrière « etc » prend tout son sens dans la théorie des ensembles et son axiomatique. En particulier, les axiomes de Peano donnent naissance aux entiers naturels.

Axiome 1 (Peano)**.** *Il existe un ensemble* $\mathbf{N}$ *contenant un élément particulier* α*, muni d'une application de succession* σ *vérifiant les trois axiomes fondateurs :*

(0) α *n'est le successeur d'aucun élément.*

$$\forall x \in \mathbf{N}, \quad \sigma(x) \neq \alpha.$$

(1) $\mathbf{N}$ *est la seule partie de* $\mathbf{N}$ *stable par* σ *et contenant* α*.*

$$\forall X \subset \mathbf{N}, \quad \alpha \in X \text{ et } \sigma(X) \subset X \Longrightarrow X = \mathbf{N}.$$

(2) σ *est injective.*

$$\forall x, y \in \mathbf{N}, \quad \sigma(x) = \sigma(y) \Longrightarrow x = y.$$

Admettre cette axiomatique, c'est accepter le raisonnement par récurrence. Une méthode de raisonnement qui sera critiquée un certain temps et qui aboutit à la conception des langages récursifs dans les années soixante. Aujourd'hui, il est inconcevable de développer des programmes informatiques sans avoir recours d'une façon ou d'une autre aux algorithmes récursifs.

Exercice 27. *Écrire un algorithme d'addition et de multiplication à partir des trois notions : zéro, prédécesseur et successeur.*

Un ensemble est fini ou infini. Suivant Dedekind, un ensemble X est infini s'il est équipotent à l'une de ces parties propres. L'ensemble des entiers naturels est le premier exemple d'ensemble infini. L'argument diagonal montre que l'ensemble des nombres réels est immensément plus important. L'un est uns puits sans fin, et l'autre une mer infiniment large mais peu profonde. Une image que proposait mon prof de logique Fraissé. Cantor croyait qu'il n'y avait pas d'ensembles infinis strictement compris entre $\mathbf{N}$ et $\mathbf{R}$. Mais en 1963, Cohen à démontré que cette question est indécidable. On peut admettre ou pas l'axiome du continu.

Axiome 2 (continu). *Tout ensemble infini strictement subpotent à l'ensemble des nombres réels est dénombrable.*

16. ΕΥΚΛΕΙΔΟΥ : ΣΤΟΙΧΕΙΟΝ Ζ΄

À l'image de la trigonométrie, les travaux des géomètres grecs traversent les siècles sans difficultés tant les applications sont nombreuses et importantes. Pensez par exemple à la construction des monuments, pyramides, châteaux et autres cathédrales ! D'un autre côté, l'arithmétique beaucoup trop fondamentale connaît de très sérieuses difficultés...

Au III-ième siècle av J.-C., Euclide synthétise les connaissances grecques dans ses éléments, une oeuvre composée de 13 tomes. Les tomes 1 à 6 n'auront pas trop de mal à survivre alors que le livre 7 consacré aux nombres en vient presque à disparaître à cause des guerres mais aussi du manque d'intérêt de l'humanité envers les nombres. Encore aujourd'hui, les historiens n'arrivent pas à localiser précisément dans le temps l'oeuvre arithmétique de Diophante, entre -200 et $+200$.

Pour les grecs, la notion de nombre est systématiquement associée à la notion de mesure. Ils évitent de donner le statut de nombre à zéro mais aussi à un. Deux mesures sont commensurables s'il existe une unité pour les mesurer. Le « septième élément » d'Euclide commence par préciser la notion de nombre, ci-dessous j'ai rapporté 8 de ses 18 définitions et postulats.

1 L'*unité* est ce relativement à quoi tout objet est appelé *Un*.
2 Un *nombre* est une collection d'unités.
3 Un nombre es appelé *partie* d'un nombre plus grand quand il divise exactement ce dernier.
5 Quand un nombre est divisible par un autre plus petit que lui, on l'appelle *multiple* de ce dernier.
6 Tout nombre divisible par 2, est un nombre *pair*.
7 Tout nombre non divisible par 2 où différent d'une unité d'un nombre pair, est un nombre *impair*.
12 On appelle *nombre premier*, tout nombre ayant comme seul diviseur l'unité.
13 Deux ou plusieurs nombres sont *premiers entre-eux*, s'ils n'ont pas d'autre diviseur commun que l'unité.

Lorsqu'étant donnés deux nombres inégaux AB et $\Gamma\Delta$ (avec $A > \Gamma\Delta$), en retranchant successivement le plus petit $\Gamma\Delta$ du plus grand AB, et autant de fois que cela est possible, jusqu'à ce que l'on arrive à une reste AZ inférieur à $\Gamma\Delta$ et en continuant ainsi (en retranchant AZ de $\Gamma\Delta$ etc), il n'arrive jamais qu'un reste divise sont précédent, jusqu'à ce que l'on obtienne un reste égal à l'unité, alors les deux nombres AB et $\Gamma\Delta$ sont premiers entre-eux.

Après quoi Euclide explique la méthode bien connue des commerçants grecs pour retrouver le plus grand diviseur commun de deux nombres non premiers entre-eux.

Soit AB et $\Gamma\Delta$ deux nombres non premiers entre-eux, dont on doit chercher le PGCD.
(1) Si le nombre $\Gamma\Delta$ divise le nombre AB, étant aussi diviseur de lui même, il est donc diviseur commun de AB et $\Gamma\Delta$. Et il est évident qu'il est leur PGCD, *car aucun nombre plus grand que $\Gamma\Delta$ ne peut être son diviseur.*
(2) Si $\Gamma\Delta$ ne divise pas AB, en appliquant la méthode des soustractions successives, il y aura un reste qui divise son précédent. Il est en effet que ce reste soit l'unité, car dans ce cas les nombres donnés seraient premiers entre-eux, contrairement à l'hypothèse.
Supposons alors que $\Gamma\Delta$ mesurant BE, laisse un reste $EA < \Gamma\Delta$ et que EA mesurant ΓZ, laisse un reste $Z\Gamma < EA$ et que ΓZ mesure exactement AE. Puisque ΓZ mesure BE et par conséquent ΓZ mesure BE ; et il mesure aussi EA. Il mesure par conséquent le nombre BA tout entier ; comme il mesure aussi $\Gamma\Delta$, il s'en suit que ΓZ mesure AB et $\Gamma\Delta$. C'est donc leur diviseur commun. Je dis maintenant qu'il est leur PGCD.
En effet, dans le cas contraire, un nombre $H > \Gamma Z$ serait diviseur commun de AB, $\Gamma\Delta$. Comme $\Gamma\Delta$ mesure

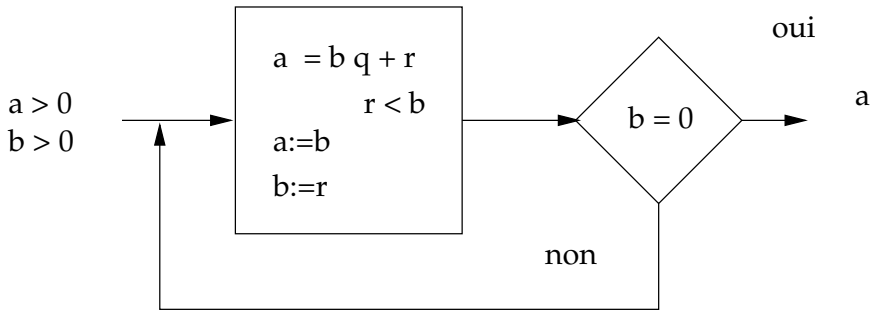


FIG. 4. Calcul de PGCD(a, b).

BE, ce nombre H serait diviseur de BE et de BA, et par conséquent aussi du reste AE ; et comme AE divise ΓZ, le nombre H sera diviseur de ΔZ. Mais H mesure ΔΓ et par conséquent il mesure le reste ΓZ. C'est-à-dire qu'un nombre serait diviseur d'un autre nombre plus petit ; ce qui est impossible.
Il n'est donc pas possible qu'un nombre plus grand que ΓZ soit diviseur commun de AB et ΓΔ. Donc ΓZ est le PGCD *des nombres AB et ΓΔ.*

Les procédures TEX calculent le PGCD de deux entiers par la méthode d'Euclide. Elles illustrent les mécanismes itératif et récursif du langage.

```
\def\PgcdIteratif{
    \loop
       \ifnum\a<\b \x=\b \b=\a \a=\x\fi
       [\number\a,\number\b],
       \advance\a by -\b
     \ifnum\b>0
     \repeat
}
```

```
\def\PgcdRecursif{
    \ifnum\a<\b \x=\b \b=\a \a=\x\fi
    [\number\a,\number\b],
    \ifnum\b=0 \let\next=\relax
         \else\advance\a by -\b
              \let\next=\PgcdRecursif\fi
     \next
}
```

Proposition 5 (Lemme d'Euclide)**.** *Soient a et b deux nombres entiers premiers avec un troisième c. Le produit ab est premier avec c.*

Le lemme d'Euclide est vrai dans tous les anneaux factoriels, c'est-à-dire les anneaux dans lesquels la décomposition en facteurs irréductibles existe avec unicité. Voici un contre-exemple instructif. Considérons le sous-anneau A du corps des nombres complexes composé des éléments qui s'écrivent sous la forme $a+bi\sqrt{5}$. Il s'agit bien d'un sous-anneau, vérifiez !

a. Montrer que la norme envoie A dans l'ensemble des entiers naturels.
b. Montrez que 2 et 3 sont irréductibles dans A.
c. Montrez que $1 + i\sqrt{5}$ est irréductible.
d. idem pour le conjugué.
e. Montrez que 2 et $1 + i\sqrt{5}$ sont étrangers.
f. idem avec le conjugué.
g. Vérifiez que $2 \times 3 = (1 + i\sqrt{5})(1 - i\sqrt{5})$.
h. Alors, surpris ?

Exercice 28. *Démontrer l'irrationalité de* $\sqrt{2}$.

Proposition 6 (Algorithme d'Euclide)**.** *Soient a et b deux nombres entiers, b non nul. le* PGCD *des entiers a et b est déterminé en nombre fini d'étapes par le procédé décrit dans le schéma 4.*

Nous utiliserons la logique de Hoare pour démontrer l'affirmation d'Euclide.

17. Reduction modulaire

```
REDUCTION( a, b: REGISTRE)
//reduction de a modulo b
VARIABLE   q : CHIFFRE;
           r : REGISTRE;
       i, j : INDICE;
DEBUT
  j := DEGRE(b);
  i := TAILLE(a) - 1;
  ZERO(r);
  TANTQUE ( i >= j)
      B-MUL( r );
      r[0] := a[i];
      q := ESTIMATION(r,b);
      XSTS( r, q, b);
      TANTQUE(CMP(r,b)>= 0)
          STS(a, b)
          q := q - 1;
      FINTQ
      i := i - 1;
  FINTQ
FIN
```

Soient a et b deux entiers naturels, b non-nul. Effectuer la division Euclidienne de a par b c'est déterminer le quotient q et le reste r, deux autres entiers satisfaisant aux deux conditions $a = bq+r$ et $0 \leq r < b$. Diviser a par b c'est calculer le quotient q alors que réduire a modulo b c'est calculer le reste r.

Dans un système de numération de base B, les entiers sont représentés par une suite de chiffres. On écrit par exemple que $a = (a_{n-1} \dots a_1 a_0)$ pour exprimer que les chiffres a_i sont des entiers positifs ou nuls, inférieurs à B et tels que $a = a_{n-1}B^{n-1} + \dots + a_1 B^1 + a_0$.

L'algorithme de réduction modulaire proposé ci, utilise les opérations sur les chiffres pour déterminer les chiffres du quotient et du reste. Les chiffres du quotient sont déterminés du plus significatif au moins significatif comme nous l'avons appris à l'école primaire, les restes intermédiaires sont stockés dans le registre r. La fonction `Estimation`(a, b) retourne une approximation par défaut du quotient de a par b qui (sans finesse) peut toujours être choisie égale à 0, de sorte à obtenir un algorithme de complexité $\mathrm{O}(Bn^2)$. Les deux faits suivants permettront de donner une estimation par défaut à deux unités près.

Fait 6. *Soient a et b deux entiers non nuls. On suppose que b s'écrit sur n chiffres et que $b < a < Bb$. Notons q le quotient de a par b et $\hat{q}$ le quotient de la partie significative de a par celle de b c'est le quotient de $\alpha := (a_n a_{n-1})_B$ par $\beta := b_{n-1}$. On a :*

$$q \leq a/b \leq \hat{q}$$

De plus, si la partie significative de B est supérieure ou égale à $B/2$ alors l'estimation est presque idéale :

$$\hat{q} - 2 \leq a/b \leq \hat{q}$$

Démonstration. Pour le premier point est assez délicat, nous suivons pa preuve proposée par Knuth. De l'inégalité stricte $\hat{q} \leq \alpha/\beta < \hat{q}+1$, nous tirons l'inégalité large $\alpha \leq \beta\hat{q}+\beta-1$.

$$\begin{aligned} a - b\hat{q} &\leq a - \beta B^{n-1}\hat{q} \\ &\leq a - \alpha B^{n-1} + \beta B^{n-1} - B^{n-1} \\ &\leq \beta B^{n-1} \\ &\leq b \end{aligned}$$

Pour le second point, il suffit d'écrire :

$$\hat{q} - q = \frac{\alpha}{\beta} - \frac{a}{b} = \frac{\alpha b - a\beta}{\beta b} \leq \frac{\alpha b}{\beta b} < \frac{2\alpha}{B}$$

□

Exercice 29. *Soit z un nombre et c un chiffre. Montrer que la multiplication de z par c propage une retenue inférieure ou égale à $c - 1$.*

Exercice 30. *Soit z un nombre dont le chiffre significatif est inférieur ou égal à $B/2$. Montrer qu'il existe un chiffre c tel que le chiffre significatif de zc soit supérieur à $B/2$ tout en étant de même taille que z.*

FIG. 5. Les origines du mot algorithme.

Exercice 31. *Déduire de l'exercice précédent un algorithme de réduction modulaire plus performant. Vérifier que votre algorithme calcule la réduction modulaire de a par b $\mathrm{O}(n^2)$ opérations sur les chiffres !*

18. LÉONARD DE PISE ET AL-KHAREZMI

Les organigrammes pour décrire des processus de calculs étaient très à la mode dans les années 70–80. Sans vraiment comprendre pourquoi, je constate qu'ils ont tendance à disparaître des enseignements d'informatique laissant le champ libre aux langages et algorithmes. Algorithme ? une terminologie qui est restée longtemps mystérieuse. Pendant longtemps, on croyait devoir lire dans le mot algorithme le radical grec *arithmos* qui signifie *nombre*. La poursuite étymologique s'enlise à moins de croire en une certaine *douleur des nombres*. en effet, seul le mot grec *algos* (douleur) admet le radical adéquat ! Possible pour les étudiants de premier cycle qui patogent avec les logarithmes [8] mais pas pour les arithméticiens grecs ! Le mathématicien italien du XI-ième siècle, Léonard de Pise, rapporte de ses voyages méditerranéens le *sifr* de Perse. Lecteur des textes mathématiques arabes, Léonard de Pise est à l'origine du mot algorithme. Un terme qu'il aurait utilisé par pour décrire les procédés de calculs qu'il peut lire dans le traité *Kitab al'jabr w'al-muqabala* (règles de restaurations et réductions) écrit par un certain Al-Kwarezmi mathématicien arabe du IX-ième siècle dont le patronyme exact est : Abu Ja'far Mohammed ibn Mûsâ al-Kwarizmi, père de Jafar, Mohammed, fils de Moïse et natif d'al-Kwarezmi.

Le Khôresme, ex-Khanat de Khiva, situé sur la partie inférieure du fleuve Amou-Syria est devenue une petite ville de l'Ouzbékistan du côté de la mer d'Aral. Fibonacci s'intéresse à la résolution de l'équation du second degré, et ses successeurs de l'école italienne se distingueront dans la résolution des équations algébriques de degré supérieures. Tartaglia invente les nombres imaginaires pour donner la formule de l'équation du troisième degré, Ferreri donne celle du quatrième. Au début du XIX-ième siècle, Abel démontre l'impossibilité de résoudre radicaux les équations de degrés supérieur ou égal à 5. Un résultat souvent attribué (à tort) à Galois qui n'en reste pas moins l'inventeur de la théorie qui porte son nom.

[8] Quel superbe exemple d'anagramme !

19. Les lapins de Fibonacci

Comme vous le savez, les italiens possèdent souvent deux noms, le sobriquet de Léonard de Pise est Fibonacci. Avez vous déjà entendu parler de la prolifération des lapins de Fibonacci ?

Problème 1. *Jean de Florette étudie la rentabilité de l'élevage des lapins. Il part du principe qu'un couple de lapins donne naissance à un couple de lapins. Il sait qu'un lapin se reproduit dès l'age de* 1 *mois. Il se demande quelle est la taille de la descendance d'un couple de lapins au bout d'un an. Donnez lui un coup de main !*

Notons F_n le nombre de lapins reproductifs au n-ème mois. Convenons de poser $F_0 = 0$, puis $F_1 = 1$. Au n-ème mois, les lapins de la génération $n-2$ deviennent reproductifs, ils s'ajoutent aux F_{n-1} couples reproductifs de la génération précédente :

$$F_0 = 1, \quad F_1 = 1, \quad F_n = F_{n-1} + F_{n-2}.$$

Ce qui donne :

n	0	1	2	3	4	5	6	7	8	9	10	11	12
F_n	0	1	1	2	3	5	8	13	21	34	55	89	154

Pour savoir comment évolue F_n, on pose la relation de récurrence :

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n = \begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix}$$

Une relation simple mais très pratique.

Exercice 32. *Montrez que :*

$$(-1)^n = F_{n+1}F_{n-1} - F_n^2$$

Le polynôme caractéristique de la matrice $\left(\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ vaut $X^2 - X - 1$, dont les racines sont le nombre d'or de Phidias ϕ et son conjugué algébrique $\hat{\phi}$. On déduit d'un résultat d'algèbre linéaire que F_n est une combinaison linéaire de ϕ^n et $\hat{\phi}^n$.

Exercice 33. *Démontrez (au pire par induction sur n) que :*

$$F_n = \frac{1}{\sqrt{5}}(\phi^n - \hat{\phi}^n) \tag{2}$$

20. Le siècle des Lumières

MDC, la dernière année du XVI-ième siècle est marquée par un des évènements les plus triste de l'histoire des sciences. Giordano Bruno promoteur des idées du plus prudent Coppernic, est convoqué par l'inquisition. Lors de son jugement, il déclare sa foi, sa croyance en Dieu mais persiste à dire que selon lui l'univers est infini et que la Terre tourne autour du Soleil. Une conviction qui le conduit tout droit au bûcher le 16 Juillet 1600. L'année d'après s'ouvre le *siècle des Lumières*, au sens propre et au sens figuré. De grandes villes fondent des lieux d'échanges, et de disputes : des universités, les idées s'y déversent à flots et tourbillonnent. Avec toute cette effervescence, pour ne pas être de reste l'Église de Rome fonde sa propre université qui contribue à la formation d'un scientifique de premier plan, Gallilei Gallileo.

La science est encore régie par les principes du grec Aristote quand Gallilée invente l'expérimentation numérique. Il s'agit convertir les expériences en nombres, dans le but de remplacer les explications philosophiques par des faits mathématiques. C'est ainsi qu'il découvre des lois de la cinématique. Par ailleurs, Gallilée met au point une lunette performante, il découvre deux petits astres qui gravitent autour de Jupiter. Des objets célestes qui ne tournent pas autour de la Terre ! Sur la base de cette observation, il admet intuitivement la thèse de Coppernic et Bruno. La thèse héliocentrique n'est pas nouvelle

puisqu'elle est proposée par le grec Aristarque de Samos vers 350 avjc. Quoiqu'il en soit, elle demeure révolutionnaire et l'avis de Gallileo se répand dans l'Europe entière.

Sûr de lui, Gallilée entreprend de faire changer d'avis l'Église. Les intellectuels de l'époque, catholiques compris, adhèrent à cette idée. Toutes les conditions semblent être réunies pour rendre raison à Bruno mais Gallilée se veut l'inventeur de cette découverte, il ignore copieusement les progrès du mathématicien Keppler. Vanité, intransigeance de Gallileo ajoutées à l'autoritarisme d'une Église en quête d'une opération coup de poing emporteront Galliée par deux fois devant les inquisiteurs. En 1633, Gallileo doit se rétracter à jamais... De son côté, Keppler définit le modèle héliocentrique. Quelques décennies plus tard, Isaac Newton et sa loi de gravitation universelle fourniront des explications.

Monseigneur Lemaitre initiateur du « big bang » fait beaucoup plus que défendre la thèse de Bruno, ses idées délirantes conduisent à la réconciliation de l'Église et des sciences. Le 31 Octobre 1992, le pape Jean-Paul II réhabilite Gallei Gallileo. Comme dirait Évariste Galois :« Quel gâchis ! »

21. Interdisciplinarité

Nous le savons aujourd'hui, l'introduction des nombres en physique est le point de départ de progrès théoriques et technologiques formidables. L'union fait la force, et quand une discipline piétine, elle commence à observer les autres. Une attitude qui provoque des rencontres dans lesquelles s'échangent quelques petits trucs et astuces pour aller plus loin. La tragédie de la section précédente décrit un processus d'interdisciplinarité involontaire qu'il convient de dédramatiser...

Dieu dit « que la lumière soit » et

$$-\frac{\partial \vec{D}}{\partial t} + \operatorname{rot} \vec{H} = J \qquad \frac{\partial \vec{B}}{\partial t} + \operatorname{rot} \vec{E} = 0$$
$$\operatorname{div} \vec{B} = 0 \qquad \operatorname{div} \vec{D} = \rho$$

furent...

A physicist, an engineer, and a computer scientist were discussing the nature of God. Surely a Physicist, said the physicist, because early in the Creation, God made Light ; and you know, Maxwell's equations, the dual nature of electro-magnetic waves, the relativist consequences... An Engineer !, said the engineer, because before making Light, God split the Chaos into Land and Water ; it takes a hell of an engineer to handle that big amount of mud, and orderly separation of solids from liquids... The computer scientist shouted : And the Chaos, where do you think it was coming from, hmm ?

—Anonymous, somewhere on the web.

L'interdisciplinarité fait peur, comme toujours dans ce genre d'affaire, on voit bien ce qu'il y a à perdre mais pas ce qu'il y a à gagner. Pourtant c'est clair, mis à part l'association souvent désastreuse de la politique et des sciences, tous les autres mélanges devraient être bénéfiques. Certaines disciplines sont allées très loin dans la voie de la spécialisation, et maintenant, il semble raisonnable de promettre un bel avenir aux scientifiques dotés d'un esprit d'ouverture. Pour cela, je souhaite aux apprentis chercheurs XXI-ième siècle une formation scientifique et intellectuelle qui inclue l'interdisciplinarité. Évitez les mentions et diplômes mono-disciplinaires !

La cryptographie est un bel exemple de sujet interdisciplinaire. Pour ne pas être spectateur des méthodes cryptographiques, il faut intégrer des connaissances en probabilité et analyse, une large part de l'arsenal des mathématiques discrètes, une bonne culture informatique : algorithmique et l'informatique fondamentale. L'existence de RSA ou encore des systèmes basés sur les courbes elliptiques, montre qu'une bonne pratique de la théorie des nombres est un atout. Mais il n'y a pas de limites ! Certaines notions de la physique quantique suggèrent la possibilité de construire des machines pour lesquelles un bit d'information correspond à l'état d'un électron d'un noyau d'hydrogène. Il s'agit d'un bit quantique qui peut prendre une infinité non dénombrable de valeurs ! Il en résulte une puissance de calcul infiniment plus performante que celle les ordinateurs d'aujourd'hui. Le principe d'incertitude d'Heisenberg rend difficile la programmation de ces machines, mais

la théorie des codes correcteurs donne des solutions. Quand toutes ces balivernes verront le jour, les « computers scientists » auront bien changés.

22. L'identité de Bachet

Vous avez certainement entendu parlé de l'identité de Bézout. Peut-être vous destinez vous à une carrière d'enseignant ? Dans ce cas, je vous souhaite de réussir au mieux, mais permettez moi une petite faveur : contribuez à changer le cours des choses en rendant à César ce qui lui appartient.

Proposition [Bachet] Soient a et b deux entiers naturels. Désignons par d leur pgcd. Il existe deux entiers relatifs u et v de valeurs absolues inférieures à b et a respectivement tels que $au + bv = d$. En particulier, si $\text{PGCD}(a, b) = 1$ alors u est l'inverse de a modulo b.

D'après J. Itard , l'identité est est due à Bachet. Le traducteur des oeuvres de Diophante du XVI-ième siècle et non pas à Étienne Bézout professeur à polytechnique contemporain d'Euler qui a probablement généralisé cette identité au cas des polynômes, d'où la confusion.

```
ALGORITHME EUCLIDE++(a,b,u,v)
   VALEUR     a, b : ENTIER;
   ADRESSE     u,v : ENTIER;
   RETOUR          : ENTIER;
   VARIABLE   x, y : ENTIER;
              q, r, d : ENTIER;
DEBUT
   SI ( b = 0 ) ALORS
                u := 1;
                v := 0;
                RETOURNER(a);
   FSI
   q := a DIV  b;
   r := a MOD  b;
   d := EUCLIDE++(b,r, x,y);
   u := y;
   v := x - q*y;
   RETOURNER(d);
FIN
```

Exercice 34. *Utiliser l'algorithme* **`Euclide++( )`** *pour démontrer ... l'identité de Bachet !*

23. Complexité de l'algorithme d'Euclide

La question du nombre d'itérations de l'algorithme d'Euclide ou de la version étendue à été résolue par Gabriel Lamé dans la première partie du XIX-ième siècle. Elle repose sur un lemme.

Lemme 1. *Soient $0 \leq b < a$ deux entiers. Si* **`Euclide`**(a, b) *utilise k itérations pour retourner d alors*

$$dF_k \leq b \quad et \quad dF_{k+1} \leq a$$

Démonstration. Il suffit de raisonner par induction. Si l'algorithme répond sans itérer, c'est que $b = 0$ et $a >= 0$. Dans ce cas, nous avons bien $dF_0 = 0 \leq b$ et $dF_1 = d = a$. De proche en proche, si l'algorithme itère k fois c'est que $a = bq + r$ et `Euclide(b, r)` itère $k - 1$ fois pour retourner d.

$$dF_k \leq r \quad \text{et} \quad dF_{1+k} \leq b$$

et donc $dF_k + dF_{1+k} = dF_{2+k} \leq b + r \leq a$.

□

Théorème 3 (Lamé)**.** *Soit $k \geq 1$ un entier et soient a et b deux entiers tels que $0 \leq b < a$ et $a < F_{k+1}$. Alors, l'algorithme d'Euclide utilise au plus k itérations. La complexité de l'algorithme d'Euclide est logarithmique sur les entiers et $\mathrm{O}(\log(a)^3)$ sur les grands-entiers.*

Démonstration. Il suffit de se rappeler que $k = \Theta(\log(F_k))$. □

En utilisant le relation $F_{m+n} = F_{n+1}F_m + F_nF_{m+1}$, Lucas remarquera un peu plus tard que le résultat de Lamé est optimal grâce à la relation

$$\text{PGCD}(F_m, F_n) = F_{\text{PGCD}(m,n)} \tag{3}$$

24. Aspect logique de l'algorithme d'Euclide

```
ALGORITHME EUCLIDE(a, b )
   VALEUR     a, b : ENTIER;
   VARIABLE   q, r : ENTIER;
DEBUT
   TANT QUE ( b > 0 ) FAIRE
   (q, r) := DIVISION(a,b);
        a := b;
        b := r;
   FTQ
   RETOURNER(a);
FIN
```

Sans récursivité, l'algorithme d'Euclide est souvent choisi pour illustrer le fonctionnement de la logique de Hoare. Vous êtes convaincu du caractère théorique de cette logique, alors vous devez savoir que le tri rapide version QuickSort a été inventé par Hoare : C. A. R. Hoare, un autre *computer scientist* ! Commençons par remarquer que l'algorithme s'arrête : b décroît et reste positif, donc... c'est encore un argument de descente infinie ! Les deux propositions logiques $0 \leq b$ et $\delta|a \wedge \delta|b$ sont invariants par l'unique boucle de l'algorithme `EUCLIDE(a,b)`. Démontrons le pour la seconde. Dans un style « boustrophidon », nous écrivons :

$$\begin{array}{ccccc}
(\delta|a)\wedge(\delta|b) & \xrightarrow{(q,r):=division(a,b)} & (a=bq+r)\wedge(\delta|a)\wedge(\delta|b) & & \\
 & & \downarrow & & \\
(\delta|r)\wedge(\delta|b) & \longleftarrow & (\delta|r)\wedge(\delta|a)\wedge(\delta|b) & & \\
\downarrow{\scriptstyle a:=b} & & & & \\
(\delta|r)\wedge(\delta|a) & \xrightarrow{b:=r} & (\delta|b)\wedge(\delta|a) & &
\end{array}$$

Il suit que la valeur de $\text{PGCD}(a,b)$ est invariante. En sortie de boucle, nous récupérons la condition d'arrêt $(b \leq 0)$ et l'invariant $(0 \leq b)$ ce qui implique $b = 0$ et donc $\text{PGCD}(a,b) = \text{PGCD}(a,0) = a$.

De même, on peut « dérécursifier » l'algorithme

`EUCLIDE++(a,b,u,v)`

en introduisant la matrice $\left(\begin{smallmatrix} 1 & 0 & a \\ 0 & 1 & b \end{smallmatrix}\right)$. Les combinaisons linéaires appliquées aux entiers a et b sont appliquées aux vecteurs lignes et si l'une des lignes vaut (x,y,z) alors une égalité du type $\alpha x + \beta y = z$ reste invariante. La suppression de toutes les variables tableaux conduit à un algorithme efficace : c'est l'algorithme de Blanksenship.

25. Epilogue

Vous le savez bien, l'obsession du théoricien est la généralisation. Généraliser c'est s'élever pour mieux voir, sortir les mains du cambouis ! La problématique qui est au coeur de notre histoire est celle du PGCD. Du point de vue géométrique, il s'agit à partir de deux vecteurs $\vec{a}$ et $\vec{b}$ portés par une même droite, de déterminer une combinaison linéaire à coefficients entiers qui soit de longueurs minimale sans être nulle.

Mais que se passe-t-il si nous supposons les vecteurs $\vec{a}$ et $\vec{b}$ non colinéaires ? Dans cette première généralisation étudiée par C. F. Gauss, il s'agit de trouver le vecteur le plus court dans un réseau plan engendré par deux vecteurs.

Généralisons encore ! Comment déterminer un vecteur court non nul dans un réseau de dimension arbitraire ? La solution algorithmique LLL proposée par Lovasz, Lenstra père et fils est devenue très populaire ces derniers temps. En effet, les implantations soignées de l'algorithme LL sont des outils efficaces pour casser un bon nombre de cryptosystèmes.

La boucle est fermée, nous venons de retrouver une devise chère au directeur du notre laboratoire : *la théorie c'est pratique !* [9]

In conclusion, let me encourage all of you to strive for a healthy balance between theory and practice in your own lives. If you find that you're spending almost all your time on theory, start turning some attention to practical things ; it will improve your theories. If you find that you're spending almost all your time on practice, start turning some attention to theoretical things ; it will improve your practice.

D.E.K.

Références

[1] Allegre C. *Dieu face à la science.* Fayard, 1997.

[2] Bombieri E. Problems of the millenium : the riemann hypothesis. *www.claymath.org*, 2000.

[3] Chabert Jean-Luc & als. *Histoire d'algorithmes.* 1999.

[4] Demazure Michel. *Cours d'algèbre :primalité, divisibilité, codes.* Cassini, 1997.

[5] Ebbinghaus H.-D. & als. *Les Nombres...* Vuibert, 1998.

[6] Gauss C. F. *Disquisitiones Arithmeticae*, volume section VII. 1807.

[7] Itard J. *Arithmétique et Théorie des Nombres*, volume 1093 of *Que sais-je ?* PUF, 1963.

[8] Itard J. *Les nombres premiers*, volume x of *Que sais-je ?* PUF, 1975.

[9] Kayas Georges. *Euclide, les éléments.* Edition du CNRS, 1978.

[10] Knuth D. *The Art of Computer Programming*, volume 1. Addison Wesley, 1973.

[11] Knuth D. *The Art of Computer Programming*, volume 2. Addison Wesley, 1973.

[12] Knuth D. *The Art of Computer Programming*, volume 3. Addison Wesley, 1973.

[13] Knuth D. *The TEX book.* Addison Wesley, 1984.

[14] Koblitz N. *A course in Number Theory and Cryptography*, volume 114. Springer-Verlag, 1991.

[15] Koblitz Neal. *A Course in Number Theory and Cryptography.* Springer-Verlag, 1994.

[16] Seroul R. *Le petit livre de TEX.* InterEditions, 1989.

[17] Seroul R. *math-info, informatique pour mathématiciens.* InterEditions, 1995.

[18] Wilf S. *Algorithms and Complexity.* Prentice-Hall International, 1986.

[9] Mais il ne faut jamais cesser d'en fournir une démonstration.

26. Travaux Pratiques et Dirigés

Exercice 1. *La meilleure méthode connue à ce jour pour factoriser un enitier n est due à Pomerance (1980), elle est de complexité* $O(e^{\sqrt{\log n \log\log n}})$. *Le* 22 *Août* 1999, *un entier* RSA *de* 155 *chiffres décimaux a été factorisé en* 4 *mois par* 300 *machines ce qui équivaut à un travail de* 8000 *MIPS-année. Estimez le temps de calcul de factorisation d'un entier* RSA *de* 200 *chiffres.*

Exercice 2. *Comment implantez vous la structure de registre en langage C ? Écrire un algorithme de multiplication d'un registre par un chiffre pour une base inférieure à* $2^{32}-1$.

Exercice 3. *Soit z un nombre et c un chiffre. Montrez que la multiplication de z par c propage une retenue inérieure ou égale à* $c-1$.

Exercice 4. *Soit z un nombre dont le chiffre significatif est inférieur ou égal à* $B/2$. *Montrez qu'il existe un chiffre c tel que le chiffre significatif de cz soit supérieur à* $B/2$ *et tel que cz et z soient de même taille.*

Exercice 5. *Donnez une estimation du nombre de chiffres du nombre* $1000!+1$ *en base* 10, *en base* 10000. *Comment passer de la base* 10000 *à la base* 10 *? En déduire un algorithme efficace pour calculer* 1000! *en base* 10.

algorithmes	action	complexité
`INIT(x : REGISTRE ; c :CHIFFRE)`	$x := c$	$\Theta(N)$
`TEST(x : REGISTRE)`	$x = 0$ $x = 1$?	$\Theta(1)$ (moyenne)
`COPIER(x,y : REG.)`	$x := y$	$\Theta(1)$ (moyenne)
`CMP(x, y : REG.)`	$x = y$?	$\Theta(N)$
`INC(x :REGISTRE)`	$x := x + 1$	$\Theta(1)$ (moyenne)
`ADD(x,y :REGISTRE)`	$x := x + y$	$\Theta(N)$
`STS(x,y :REGISTRE)`	$x := x - y$	$\Theta(N)$
`C-MUL(x :REGISTRE, c :CHIFFRE)`	$x := cx$	$\Theta(N)$
`DICHO(x :REGISTRE)`	$x := x/2$	$\Theta(N)$
`B-MUL(x :REGISTRE)`	$x := Bx$	$\Theta(N)$
`XADD(x,y :REGISTRE, c :CHIFFRE)`	$x := x + cy$	$\Theta(N)$
`XSTS(x,y :REGISTRE, c :CHIFFRE)`	$x := x - cy$	$\Theta(N)$

Exercice 6. *Utilisez les primitives décrites dans la table pour pour obtenir un algorithme* ***`PRD(x,y :REGISTRE)`*** *qui calcule le produit de x par y. Quelle est sa complexité ?*

Exercice 7. *Utilisez les primitives décrites dans la table pour pour obtenir un algorithme* ***`REDUCTION(x,y :REGISTRE)`*** *qui calcule la réduction de x modulo y. Quelle est sa complexité ?*

Exercice 8. *Soit p un nombre premier et k un entier. Montrer que si* $0 < k < p$ *alors le coefficient binomial* $C(k,p)$ *est divisible par p. Déduisez que pour tout entier x, on a :* $(x+1)^p \equiv x^p \pmod p$. *Démontrez le petit théorème de Fermat.*

Exercice 9. *Implantez la fonction* ***`TEMOIN`****(a,r) qui renvoie VRAI si le chiffre a est un témoin de Fermat du registre R et FAUX sinon. Quelle est la complexité de votre algorithme ? Utilisez votre programme pour tester la primalité des nombres* $n!+1$, *pour n au plus égal à* 100.

Exercice 10. *L'algorithme* ***`PRODUIT`****(x,y) calcule le produit de x par y. Estimez la valeur maximale que peut prendre une cellule du tableau z. Dans une implantation en langage C, on représente les chiffres par des* ***`unsigned int`***, *pour travailler en base* 256. *Quelle valeur ne doit pas dépasser* ***`max`*** *?*

Exercice 11. *Dans le cours, nous avons eu l'occasion de citer quelques illustres personnages : Archimède, Aristote, Diophante, Eratostène, Euclide, Hépitarque, Platon, Pythagore, Thalès etc...Replacez ces personnages dans le temps.*

Exercice 12. *Utiliser la méthode du crible d'Eratostène pour compter le nombre de premiers inférieurs ou égaux à n. Il s'agit d'écrire un algorithme. Suggérez quelques améliorations. Vérifiez que la complexité de votre algorithme est $O(n \log n)$.*

```
FONCTION PRODUIT( x, y : REGISTRE)
VARIABLE i : INDICE;
         z : REGISTRE
       tmp : CHIFFRE;
DEBUT
i := 0;
TANTQUE (i < MAX)
    j = 0;
    TANTQUE (i+j < MAX)
       z[i+j] = z[i+j] + x[i]*y[j];
       INC(j);
    FINTQ
    INC(i);
FINTQ
  i := 0;
ret := 0;
TANTQUE ( i < MAX)
    x[i] := z[i] + ret;
    ret  := x[i] DIV B
    x[i] := x[i] MOD B;
FINTQ
```

```
FONCTION CRIBLE(n : ENTIER)
VARIABLE i : INDICE;
         p : TABLEAU[n] BOOLEEN;
       cpt : ENTIER;
DEBUT
i := 0;
TANTQUE (i < MAX)
    P[i] = VRAI;
    INC(i);
FINTQ
cpt := 0;
i   := 0;
TANTQUE ( i < MAX)
    SI ( P[i]) ALORS
       j := 2*i;
       INC(cpt);
       TANTQUE (j<MAX)
           p[j] := FAUX;
           j := j + i;
       FINTQ
    FSI
    INC(i);
FINTQ
RETOURNER(cpt)
```