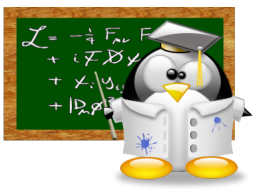
Sommaire



Prérequis

- Raisonnement par récurrence
- Fonction logarithme népérien

Définitions *Divisibilité*

Soient a et b deux nombres relatifs.
On dit que a `divise` b, et on note $a|b$, ou que b est un `multiple` de a, s'il existe un entier naturel q tel que $b = qa$. q est alors appelé le `quotient` de b par a.

Propriété

Soient a et b deux nombres relatifs.
Si un nombre c divise a et b, alors il divise tout nombre de la forme $au + bv$, où $(u\,;v) \in \mathbb{Z} \times \mathbb{Z}$.
En particulier, $c|(a-b)$ et $c|(a+b)$.

Démonstration

$$\left.\begin{array}{l} c|a \Rightarrow \exists q \in \mathbb{Z},\ a = qc \\ c|b \Rightarrow \exists q' \in \mathbb{Z},\ b = q'c \end{array}\right\} \Rightarrow \forall\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ au + bv = c(qu + q'v).$$ Donc $c|(au+bv)$. ■

Les propriétés suivantes seront admises :

Propriétés

Quels que soient les nombres relatifs a, b et c,

1. $a|a$
2. $c|b$ et $b|a \Rightarrow c|a$
3. $a|b$ et $b|a \Rightarrow |a| = |b|$
4. $ac|ab$ et $a \neq 0 \Rightarrow c|b$
5. $1|a$
6. $a|0$
7. $0|a \Rightarrow a = 0$
8. $b|a$ et $a \neq 0 \Rightarrow 0 < |b| \leqslant |a|$

Théorème *Division euclidienne*

Soient $(a\,;b) \in \mathbb{Z} \times \mathbb{Z}^*$.
Il existe un unique couple $(q\,;r) \in \mathbb{Z} \times \mathbb{Z}$ tel que :

$$a = bq + r \quad , \quad 0 \leqslant r < |b|$$

Démonstration

- **Existence d'un tel couple.**
  Supposons que $(a\,;b) \in \mathbb{Z} \times \mathbb{N}^*$.
  Alors, il existe un entier relatif q tel que :

$$q \leqslant \frac{a}{b} < q + 1.$$

  Ainsi, comme $b > 0$:

$$bq \leqslant a < bq + b. \qquad (1)$$

…

Démonstration (suite)

Posons alors :

$$r = a - bq. \tag{2}$$

Alors,

$$\begin{array}{rcl} (1) & \Rightarrow & 0 \leqslant r < b \\ (2) & \Rightarrow & a = bq + r \end{array}$$

Il existe donc bien un couple $(q\,;r) \in \mathbb{Z} \times \mathbb{Z}$ tel que $a = bq + r$, $0 \leqslant r < b$.
Pour généraliser au cas où $b \in \mathbb{Z}^*$, on pose $b = |b|$ dans ce qui précède.

- **Unicité du couple.**
  Supposons qu'il existe deux couples $(q\,;r)$ et $(q'\,;r')$ tels que :

$$\left\{ \begin{array}{lll} a = bq + r & , & 0 \leqslant r < b \\ a = bq' + r' & , & 0 \leqslant r' < b \end{array} \right.$$

  Si $0 \leqslant r < b$ et $0 \leqslant r' < b$, alors, par soustraction, $r - r' = 0$, soit $r = r'$. Et donc, par suite, $q = q'$.

■

Théorème *Algorithme d'Euclide*

Soient a et b deux entiers naturels tels que $a = bq + r$, $0 \leqslant r < b$.
On considère alors les suites (a_n),(b_n), (q_n) et (r_n) définies par :

$$\left\{ \begin{array}{l} a_0 = a \\ b_0 = b \\ q_0 = q \\ r_0 = r \end{array} \right. , \quad \left\{ \begin{array}{l} a_{n+1} = b_n \\ b_{n+1} = r_n \\ a_{n+1} = b_{n+1}q_{n+1} + r_{n+1} \end{array} \right. , \forall n \in \mathbb{N}$$

Alors, (r_n) est une suite finie dont le dernier terme est nul. L'avant dernier terme est égal à pgcd $(a\,;\,b)$.

Démonstration

Par définition, la suite (r_n) est strictement décroissante. En effet, pour tout entier naturel n, r_n représente le reste de la division euclidienne de a_n par b_n. Ainsi, $0 < r_n < b_n$. Or, $b_n = r_{n-1}$ donc $0 \leqslant r_n < r_{n-1}$. Ainsi, $0 \leqslant r_n < r_{n-1} < r_{n-2} < \cdots < r_1 < r_0$.
Or, $r_n \in \mathbb{N}$. Ainsi, la suite (r_n) est finie et son dernier terme est 0.
D'après la première propriété du cours, si $a = bq + r$ et si $d|a$ et $d|b$, alors $d|(a - bq)$, donc $d|r$.

…

Démonstration (suite)

Ainsi, en convenant d'écrire $a_{n+1} = r_n q_n + r_{n+1}$,

$$\begin{aligned}\text{pgcd}\,(a\,;\,b) &= \text{pgcd}\,(a_0\,;\,b_0)\\ &= \text{pgcd}\,(b_0\,;\,r_0)\\ &= \text{pgcd}\,(r_0\,;\,r_1)\\ &\vdots\\ &= \text{pgcd}\,(r_{N-1}\,;\,r_N)\end{aligned}$$

en convenant de noter r_N le dernier terme NON NUL de (r_n).
Nous avons :

$$\begin{aligned}a_N &= r_{N-1}q_N + r_N\\ r_{N-1} &= r_N q_{N+1} + 0\end{aligned}$$

Alors, d'après la dernière égalité, $r_N | r_{N-1}$. Donc $\text{pgcd}\,(r_{N-1}\,;\,r_N) = r_N$.
Ainsi, $\text{pgcd}\,(a\,;\,b) = r_N$. ■

Propriété *Égalité de Bézout*

Soient $(a\,;b) \in \mathbb{Z}^* \times \mathbb{Z}^*$.
Il existe un couple $(u\,;v) \in \mathbb{Z} \times \mathbb{Z}$ tel que $au + bv = \text{pgcd}\,(a\,;\,b)$.

Démonstration *Démonstration 1*

Soient U et V deux entiers relatifs tels que $c = au + bv$ soit minimal.
Notons $d = \text{pgcd}\,(a\,;\,b)$.
$d|a$ et $d|b$ donc $d|c$; ainsi,

$$d \leqslant c \tag{3}$$

La division euclidienne de a par c s'écrit :

$$a = cq + r \quad , \quad 0 \leqslant r < c.$$

Alors,

$$r = a - cq.$$

r étant une combinaison linéaire de a et c. Or, c est une combinaison linéaire de a et b, donc r est une combinaison linéaire de a et b, tout comme c. Mais c est sensé être la plus petite combinaison linéaire de a et b possible. Donc r est nécessairement égal à 0.
Ainsi, $a = cq$ et donc $c|a$; ainsi,

$$c \leqslant d \tag{4}$$

De (3) et (4), on déduit : $au + bv = d$. ■

Démonstration *Démonstration 2*

Montrons par récurrence que dans l'algorithme d'Euclide, tous les restes s'écrivent sous la forme $r_k = au_k + bv_k$.

- **Initialisation.**
  $a = bq_0 + r_0$, $0 \leqslant r_0 < b$. Donc $r_0 = a \times 1 - bq_0$.
  r_0 s'écrit donc comme combinaison linéaire de a et b. L'initialisation est alors faite.

- **Hérédité.**
  Supposons que pour tout $i \leqslant k$, $r_k = au_k + bv_k$.

$$\begin{aligned} r_{k+1} &= a_{k+1} - b_{k+1}q_{k+1} \\ &= r_{k-1} - r_k q_{k+1} \\ &= au_{k-1} + bv_{k-1} - (au_k + bv_k)q_{k+1} \\ &= (u_{k-1} - u_k q_{k+1})\, a + (v_{k-1} - v_k q_{k+1})\, b \end{aligned}$$

  Ainsi, r_{k+1} est une combinaison linéaire de a et b.
  L'hérédité est alors vérifiée.

  Ainsi, tout reste est combinaison linéaire de a et b.
  Or, $\text{pgcd}\,(a\,;\,b)$ est le dernier reste non nul. Il est donc combinaison linéaire de a et b.

■

Méthode *Méthode (résoudre une équation de Bézout)*

La dernière démonstration nous aide à trouver une méthode pour trouver le couple $(u\,;v)$.

Prenons $a = 10\,246$ et $b = 514$. L'algorithme d'Euclide nous donne :

$$\begin{aligned} 10\,246 &= 19 \times 514 + 480 \\ 514 &= 1 \times 480 + 34 \\ 480 &= 14 \times 34 + 4 \\ 34 &= 8 \times 4 + 2 \\ 4 &= 2 \times 2 + 0 \end{aligned}$$

En remontant, on a :

$$\begin{aligned} 2 &= 34 - 8 \times 4 \\ &= 514 - 480 - 8(480 - 14 \times 34) \\ &= 514 - 9 \times 480 + 112 \times 34 \\ &= 514 - 9 \times (10\,246 - 19 \times 514) + 112(514 - 480) \\ &= -9 \times 10\,246 + 284 \times 514 - 112(10\,246 - 19 \times 514) \\ &= -121 \times 10\,246 + 2\,412 \times 514 \end{aligned}$$

On obtient alors $(u\,;v) = (-121\,;2\,412)$.

Théorème *Théorème de Bézout*

Soient a et b deux entiers naturels non nuls.
a et b sont premiers entre eux équivaut à dire qu'il existe un couple $(u\,;v) \in \mathbb{Z} \times \mathbb{Z}$ tel que $au + bv = 1$.

Démonstration

Dire que a et b sont premiers entre eux signifie que $\text{pgcd}\,(a\,;\,b) = 1$.
Donc, d'après l'égalité de Bézout,

$$\text{pgcd}\,(a\,;\,b) = 1 \Rightarrow \exists\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ au + bv = 1.$$

Réciproquement, supposons qu'il existe un couple $(u\,;v) \in \mathbb{Z} \times \mathbb{Z}$ tel que $au + bv = 1$. Notons alors $\text{pgcd}\,(a\,;\,b) = d$.
$d|a$ et $d|b$ donc $d|(au + bv)$, soit $d|1$. Ainsi, $d = 1$. ■

Théorème *Théorème de Gauss*

Soient a, b et c trois entiers naturels non nuls.
Si $a|(bc)$ et si $\text{pgcd}\,(a\,;\,b) = 1$, alors $a|c$.

Démonstration

$a|(bc) \Leftrightarrow \exists k \in \mathbb{N},\ bc = ka$.

$$\begin{aligned}
\text{pgcd}\,(a\,;\,b) = 1 &\Leftrightarrow \exists\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ au + bv = 1 \\
&\Leftrightarrow \exists\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ acu + bcv = c \\
&\Leftrightarrow \exists\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ acu + kav = c \\
&\Leftrightarrow \exists\,(u\,;v) \in \mathbb{Z} \times \mathbb{Z},\ a\underbrace{(cu + kv)}_{\in\mathbb{N}} = c \\
&\Rightarrow a|c
\end{aligned}$$

■

Propriétés

Soient a et b deux entiers naturels non nuls. On note $d = \text{pgcd}\,(a\,;\,b)$ et $m = \text{ppcm}(a\,;\,b)$.

1. $\exists\,(a'\,;b') \in \mathbb{N} \times \mathbb{N},\ \begin{cases} a = da' \\ b = db' \end{cases}$
2. Un multiple commun à a et b est un multiple de $da'b'$.
3. $m = da'b'$
4. $md = ab$

Démonstration

1. $d|a$ et $d|b$ donc on peut écrire $a = da'$ et $b = db'$.
   D'après l'égalité de Bézout,

$$\begin{aligned}\exists\,(u\,;v) \in \mathbb{Z}\times\mathbb{Z},\ au + bv = d &\Leftrightarrow da'u + da'v = d\\ &\Leftrightarrow a'u + b'v = 1\\ &\Leftrightarrow \text{pgcd}\,(a'\,;\,b') = 1\end{aligned}$$

2. Posons M un multiple commun à a et b.
   Alors, $M = \lambda a = \lambda da' = \mu b = \mu db'$.
   On en déduit que $\lambda a'|\mu b'$.et donc que $b'|\lambda a'$.
   $\text{pgcd}\,(a'\,;\,b') = 1 \Rightarrow b'|\lambda$ (théorème de Gauss) et donc $\lambda = kb'$, où $k \in \mathbb{N}$.
   On a alors $M = kb'a = kda'b'$. Donc M est un multiple de $da'b'$.
3. Considérons maintenant le nombre $da'b' = ab' = ba'$: il s'agit d'un multiple commun à a et à b. Donc, $da'b' \geqslant m$.
   Puisque m est multiple strictement positif de a et de b, on a $m \geqslant da'b'$ et on peut donc conclure $m = da'b'$.
4. En multipliant par d l'égalité de l'item 3, on obtient $md = da'db'$, soit $md = ab$.

■

Définition *Congruences*

Soient a et b deux entiers naturels non nuls.
Si $a = bq + r$, $q \in \mathbb{Z}$ et $r \in \mathbb{N}$, alors on note :

$$a \equiv b \mod q \qquad \text{ou} \qquad a \equiv b\ [q]$$

On lit : a *est* `congru` *à* b `modulo` q.

Exemples

1. $7 = 3 \times 2 + 1$ donc $7 \equiv 1 \mod 2$.
2. $205 = 20 \times 10 + 5$ donc $205 \equiv 5 \mod 10$.

Propriété

Soient a, b, c, d et n des entiers naturels non nuls.

1. $a \equiv a \mod n$.
2. Si $a \equiv b \mod n$, alors $b \equiv a \mod n$.
3. Si $a \equiv b \mod n$ et $b \equiv c \mod n$, alors $a \equiv c \mod n$.
4. Si $a \equiv b \mod n$, alors $a + c \equiv b + c \mod n$.
5. Si $a \equiv b \mod n$ et $c \equiv d \mod n$, alors $a + c \equiv b + d \mod n$.
6. Si $a \equiv b \mod n$, alors $ac \equiv bc \mod n$.
7. Si $a \equiv b \mod n$ et $c \equiv d \mod n$, alors $ac \equiv bd \mod n$.
8. Si $a \equiv b \mod n$, alors $a^k \equiv b^k \mod n$, $k \in \mathbb{N}$.

Démonstration

1. $a = 0 \times n + a$ donc on a bien $a \equiv a \mod n$.
2. $a \equiv b \mod n \Rightarrow a - b = kn,\ k \in \mathbb{N}$
   $\Rightarrow b = -kn + a,\ k \in \mathbb{N}$
   $\Rightarrow b \equiv a \mod n$.
3. $a \equiv b \mod n \Rightarrow a = kn + b,\ k \in \mathbb{N}$ et $b \equiv c \mod n \Rightarrow b = k'n + c,\ k \in \mathbb{N}$.
   Donc, $a = (k + k')n + c$, d'où $a \equiv c \mod n$.
4. Si $a \equiv b \mod n$, alors $a = kn + b$ et donc $a + c = kn + (b + c)$. D'où $a + c \equiv b + c \mod n$.
5. $a \equiv b \mod n \Rightarrow a = kn + b$ et $c \equiv d \mod n \Rightarrow c = k'n + d$.
   Ainsi, $a + c = (k + k')n + (b + d)$ et donc $a + c \equiv b + d \mod n$.
6. $a \equiv b \mod n \Rightarrow a = kn + b$, donc $ac = kcn + bc$ d'où $ac \equiv bc \mod n$.
7. $a \equiv b \mod n \Rightarrow a = kn + b$ et $c \equiv d \mod n \Rightarrow c = k'n + d$.
   Ainsi, $ac = (kn + b)(k'n + d) = [kk'n + (kd + k'b)]n + bd$ d'où $ac \equiv bd \mod n$.
8. $a \equiv b \mod n \Rightarrow a = pn + b$ donc :

$$\begin{aligned} a^k &= (pn + b)^k \\ &= \sum_{i=0}^{k} \binom{k}{i} (pn)^i\, b^{k-i} \text{ (formule du binôme de Newton)} \\ &= b^k + n \sum_{i=1}^{k} \binom{k}{i} (pn)^{i-1}\, b^{k-i} \end{aligned}$$

   Ainsi, $a^k \equiv b^k \mod n$.

■

Définition *Nombres premiers*

Soit p un entier naturel.
On dit que p est `premier` s'il n'est divisible que par 1 et lui-même.

Propriété

Soit N un entier naturel non nul et différent de 1.
Alors, soit N est un nombre premier, soit N est un produit de nombres premiers.

Démonstration

Démontrons cela par récurrence forte.

- **Initialisation.**
  Le résultat est vrai pour $N = 2$.
- **Hérédité.**
  Supposons que le résultat soit vrai pour tout entier naturel strictement inférieur à N.
  Alors, pour N, soit il est premier, soit il ne l'est pas et donc $N = kd$, $2 \leqslant k < N$ et $2 \leqslant d < N$. Par hypothèse de récurrence, k et d sont soit premiers, soit des produits de nombres premiers. Ainsi, N est un produit de nombres premiers.
  L'hérédité est alors vérifiée.

Le résultat est alors vraie. ■

Propriété *Infinitude des nombres premiers*

Soit $\mathbb{P}$ l'ensemble des nombres premiers.
Alors, $\mathbb{P}$ est infini.

Démonstration

Supposons que $\mathbb{P}$ est un ensemble fini et que :

$$\mathbb{P} = \{p_k,\ 1 \leqslant k \leqslant n\}, \qquad p_1 = 2,\ p_2 = 3,\ p_3 = 5,\ \ldots$$

Posons alors $N = p_1 p_2 p_3 \cdots p_n + 1$. Alors, $N > p_n$ donc il n'appartient pas à $\mathbb{P}$. Or, il n'est divisible par aucun des p_k, ce qui signifie qu'il est premier, ce qui est contradictoire avec l'hypothèse selon laquelle $\mathbb{P}$ est fini.
Ainsi, $\mathbb{P}$ est infini. ■

Théorème *Théorème de décomposition en produits de facteurs premiers [admis]*

Soit n un entier naturel non nul différent de 1.
Alors, n s'écrit de façon unique sous la forme :

$$n = \prod_{i=1}^{k} p_i^{\alpha_i} \qquad , \qquad \alpha_i \in \mathbb{N}^*,\ p_i \in \mathbb{P},\ i \neq j \Rightarrow p_i \neq p_j$$

Remarque

On peut aussi utiliser une notation plus pratique :

$$n = \prod_{p \in \mathbb{P}} p^{\alpha_p}$$

Propriété

Soient $a = \prod_{p \in \mathbb{P}} p^{\alpha_p}$ et $b = \prod_{p \in \mathbb{P}} p^{\beta_p}$ deux entiers naturels.
Alors,

$$\text{pgcd}\,(a\,;\,b) = \prod_{p \in \mathbb{P}} p^{\min(\alpha_p, \beta_p)} \qquad \text{et} \qquad \text{ppcm}(a\,;\,b) = \prod_{p \in \mathbb{P}} p^{\max(\alpha_p, \beta_p)}$$

Démonstration

$d = \prod_{p \in \mathbb{P}} p^{\min(\alpha_p, \beta_p)}$ divise nécessairement a et b.
De plus, $\frac{a}{d}$ et $\frac{b}{d}$ sont premiers entre eux. Donc, $d = \text{pgcd}\,(a\,;\,b)$.
D'après une propriété vue précédemment, $\text{ppcm}(a\,;\,b) = \dfrac{ab}{\text{pgcd}\,(a\,;\,b)}$. Compte tenu de l'expression précédemment démontrée, on arrive à l'expression souhaitée. ■

Exemple

Prenons $a = 2^3 \times 5^7$ et $b = 2^2 \times 3^2 \times 5^8$. Alors,

$$\text{pgcd}\,(a\,;\,b) = 2^2 \times 5^7 \qquad \text{et} \qquad \text{ppcm}(a\,;\,b) = 2^3 \times 3^2 \times 5^8$$

Propriété

Soient $n \in \mathbb{N}^* \setminus (\mathbb{P} \cup \{1\})$ et d un diviseur de n. Alors,

$$d \leqslant \sqrt{n}$$

Démonstration

Si d est un diviseur de n, alors il existe un entier k tel que $n = kd$. Supposons que $d > \sqrt{n}$ et $k > \sqrt{n}$. Alors, $kd > n$, ce qui est impossible. Donc $d \leqslant \sqrt{n}$. ■

Définition *Nombre de nombres premiers*

Le nombre de nombres premiers inférieurs ou égaux à n, $n > 1$, est noté $\pi(n)$.

Ainsi, $\pi(n) - \pi(m)$ représente le nombre de nombre premiers compris entre m et n, $m < n$. À l'aide du crible d'Ératosthène (voir page suivante), on établit que :

$$\pi(10) = 4$$
$$\pi(20) - \pi(10) = 4$$
$$\pi(30) - \pi(20) = 2$$
$$\pi(40) - \pi(30) = 2$$
$$\pi(50) - \pi(40) = 3$$
$$\pi(60) - \pi(50) = 2$$
$$\pi(70) - \pi(60) = 2$$
$$\pi(80) - \pi(70) = 3$$
$$\pi(90) - \pi(80) = 2$$
$$\pi(100) - \pi(90) = 1$$

On voit que la répartition des nombres premiers est irrégulière. De plus, on a :

n	$\pi(n)$	$\frac{n}{\ln n}$	$\left\lvert \pi(n) - \frac{n}{\ln n} \right\rvert$
10	4	4	0 %
100	25	22	12 %
10^3	168	145	15,86 %
10^4	1 229	1 086	11,64 %
10^5	9 592	8 686	9,45 %
10^6	78 498	72 382	7,79 %
10^7	664 579	620 421	6,64 %
10^8	5 761 455	5 428 681	5,78 %
10^9	50 847 534	48 254 942	5,1 %

On peut alors remarquer que la différence entre $\pi(n)$ et $\frac{n}{\ln n}$ tend à se réduire lorsque n tend vers l'infini.

On pourra alors conjecturer que :

$$\pi(n) \underset{+\infty}{\sim} \frac{n}{\ln n}$$

Complément 1: Crible d'Ératosthène

Le crible d'Ératosthène est un moyen de trouver tous les nombres premiers (en théorie).

1. On dispose dans une grille tous les nombres entiers (ici, nous allons prendre les entiers inférieurs ou égaux à 100).
2. On exclut « 1 », qui ne fait pas partie de $\mathbb{P}$.
3. On marque « 2 » comme un élément de $\mathbb{P}$, puis on noircit toutes les cases contenant les multiples de 2.
4. On marque le nombre suivant qui n'est pas noirci (ici, « 3 ») comme un élément de $\mathbb{P}$, puis on noircit toutes les cases contenant ses multiples.
5. On continue ainsi jusqu'à ce qu'il n'y ait plus de nombres non marqués.

Cela nous donne :

	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Complément 2: Nombres de Fermat

Tout part du théorème suivant :

Propriété

Si $a^m + 1$ est un nombre premier, avec $a > 1$ et $m > 1$, alors m est une puissance de 2.

Démonstration

Raisonnons par l'absurde : supposons que m ne soit pas une puissance de 2. Alors, m admet un diviseur impair $d \geqslant 3$. Dans ce cas, on peut écrire :

$$\begin{aligned} a^m + 1 &= a^{d^k} + 1 \text{ où } k \in \mathbb{N}^* \\ &= \left(a^k\right)^d - (-1)^d \\ &= \left(a^k + 1\right) f(a) \text{ (d'après la formule du binôme de Newton)} \end{aligned}$$

Ainsi, $a^m + 1$ n'est pas premier (puisqu'il est divisible par $a^k + 1$, qui n'est ni égal à 1, ni à $a^m + 1$), ce qui contredit notre hypothèse selon laquelle m n'est pas une puissance de 2. ■

Si $a^m + 1$ est premier, alors il est impair ; ainsi, a^m est pair. On en déduit alors que a est pair.
Ainsi, si on veut trouver des nombres premiers, il nous faut calculer les différentes valeurs de $a^m + 1$ où a est pair et tester leur primalité.
Or, il s'avère qu'en prenant $a = 2$, les premières valeurs sont toutes premières.

n	$2^{2^n} + 1$
0	3
1	5
2	17
3	257
4	65 537

Les `nombres de Fermat` sont les nombres $F_n = 2^{2^n} + 1$.
La conjecture de Fermat stipule que $F_n \in \mathbb{P}$. Malheureusement, ne serait-ce qu'en regardant $F_5 = 4\,294\,967\,297$, on s'aperçoit que la suite $(F_n)_{n \geqslant 0}$ croît très rapidement et qu'il est réellement difficile de tester la primalité des termes pour $n \geqslant 5$.
Mais difficile ne veut pas dire impossible car Euler a démontré que F_5 admettait un diviseur autre que 1 et lui-même.

Complément 3: Nombres premiers de Mersenne

Ici, tout part de l'étude des nombres parfaits, nombres égaux à la somme de leurs diviseurs propres.
Euclide a démontré que si $M = 2^p - 1$ est un nombre premier, alors $\frac{M(M+1)}{2} = 2^p - 1\,(2^p - 1)$ est un nombre parfait.
Plus tard, Euler a démontré que tous les nombres parfaits pairs ont cette forme (N.B. : aucun nombre parfait impair n'est connu à ce jour).

Un `nombre premier de Mersenne` est un nombre premier s'écrivant sous la forme $2^p - 1$, où p est premier.

p	$2^p - 1$
2	3
3	7
5	31
7	127

Attention : pour $p = 11$, on a : $2^{11} - 1 = 2\,047 = 23 \times 89$, donc il n'est pas un nombre premier de Mersenne.

Complément 4: Nombres de Carmichaël

Pour ces nombres, tout vient du *Petit théorème de Fermat* :

Propriété *Petit théorème de Fermat*

Si p est un nombre premier et a un entier non divisible par p, alors :

$$a^{p-1} \equiv 1 \mod p$$

Démonstration

- **Montrons d'abord que les restes des divisions euclidiennes par p de a, $2a$, ... , $(p-1)a$ sont tous distincts.**
  Nous allons raisonner par l'absurde. Supposons donc qu'il existe deux entiers k et k' dans l'intervalle $[\![1\,;p-1]\!]$ tels que les restes de ka et $k'a$ soient égaux.
  On a alors :

$$\begin{cases} ka = pq + r \\ k'a = pq' + r \end{cases}, \qquad (q\,;q') \in \mathbb{Z}\times\mathbb{Z},\ r \in [\![0\,;p-1]\!]$$

  Alors,

$$(k-k')a = p(q-q')$$

  et donc $p|(k-k')a$.
  Or, $k-k' < p$ donc $p|a$, ce qui contredit le fait que a n'est pas divisible par p. Notre hypothèse selon laquelle il existe deux restes égaux est donc fausse.
  Donc tous les restes sont distincts.
- **Fin de la démonstration.**
  De ce qui précède, on en déduit :

$$\begin{aligned} a &\equiv r_1 \mod p \\ 2a &\equiv r_2 \mod p \\ &\vdots \\ (p-1)a &\equiv r_{p-1} \mod p \end{aligned}$$

  Ainsi,

$$(p-1)\,!a^{p-1} \equiv r_1 r_2 \cdots r_{p-1} \mod p \tag{5}$$

  Or, pour $1 \leqslant i \leqslant p-1$, $r_i \in [\![0\,;p-1]\!]$ et ils sont tous distincts.
  Donc $r_1 r_2 \cdots r_{p-1} = (p-1)\,!$.
  De plus, $\text{pgcd}\,(p\,;\,(p-1)\,!) = 1$ donc la congruence (5) nous donne :

$$a^{p-1} \equiv 1 \mod p$$

■

Un `nombre de Carmichaël` est un entier composé (entier non premier différent de 0 et 1) n tel que :

$$\mathcal{P} \ :\ \forall a \in \mathbb{N},\ \text{pgcd}\,(a\,;n) = 1 \Rightarrow n|(a^{n-1}-1).$$

Le *petit théorème de Fermat* nous dit que si a est un nombre premier, alors il satisfait $\mathcal{P}$, mais la réciproque est fausse.
Les nombres de Carmichaël sont les entiers naturels qui satisfont $\mathcal{P}$ sans être premiers.

Propriété

Théorème de Korselt

Un entier positif composé n est un nombre de Carmichaël si et seulement si aucun carré de nombre premier ne divise n (on dit que n est *quadratfrei* (sans carré)) et pour chaque diviseur premier p de n, le nombre $p-1$ divise $n-1$. De plus, un tel n divise tous les $a^n - a$ (même pour a non premier à n).

Il découle de ce théorème que tous les nombres de Carmichaël sont des produits d'au moins trois nombres premiers différents.

En fait, Korselt fut le premier à observer ces propriétés, mais il n'a pas pu trouver d'exemples de nombre de Carmichaël. En 1909, Robert Daniel Carmichaël trouva le plus petit de ces nombres, 561, et ceux-ci furent nommés en son honneur.

Complément 5: Cryptographie RSA

Soit $n = pq$ un entier où $(p\,;q) \in \mathbb{P} \times \mathbb{P}$.
On pose alors $m = (p-1)(q-1)$ et on considère un nombre c premier avec m.
Enfin, on note x un entier naturel quelconque.

- **Il existe un entier d tel que $cd \equiv 1 \mod m$.**
  pgcd $(c\,;\,m) = 1$ donc, d'après le théorème de Bézout, il existe un couple $(d\,;k)$ d'entiers tel que $cd + km = 1$, et donc tel que $cd \equiv 1 \mod m$.

- **On en déduit alors que $x^{cd} \equiv x \mod p$ et $x^{cd} \equiv x \mod q$.**
  → **Si x est non divisible par p**, d'après le petit théorème de Fermat, $x^{p-1} \equiv 1 \mod p$.
  Or,

$$\begin{aligned} x^m &= x^{(p-1)(q-1)} \\ x^m &= \left(x^{p-1}\right)^{q-1} \\ x^m &\equiv 1^{q-1} \mod p \\ x^m &\equiv 1 \mod p \\ x^{km} &\equiv 1 \mod p \\ x^{km+1} &\equiv x \mod p \\ x^{cd} &\equiv x \mod p \end{aligned}$$

  → **Si x est divisible par p**, alors $x \equiv 0 \mod p$, et donc $x^{cd} \equiv 0 \mod p$.
  On peut donc bien écrire $x^{cd} \equiv x \mod p$.

  Comme p et q jouent des rôles identiques, la seconde congruence est aussi démontrée.

- **On en déduit alors que $x^{cd} \equiv x \mod n$.**
  D'après le point précédent, $x^{cd} - x$ est divisible par p et q, donc divisible par pq (car pgcd $(p\,;\,q) = 1$), soit par n. D'où la congruence.

Ce résultat nous permet alors d'établir un principe de cryptage, que l'on va appelé le `cryptage RSA` (initiales de Rivest, Shamir et Adleman qui l'ont mis au point) en 1977, utilisé notamment pour les transactions confidentielles sur Internet.

1 Pour chiffrer un message, on choisit deux nombres premiers p et q très grands, puis on calcule $n = pq$.
On pose alors $m = (p-1)(q-1)$ et on cherche deux entiers c et d tels que $cd \equiv 1 \mod m$.

2 Les messages x seront des entiers naturels dans $[\![0\,;n-1]\!]$.
Le codage de x consiste à calculer $C(x) \equiv x^c \mod n$.
Le décodage de x consiste à calculer $D(y) \equiv y^d \mod n$.
On a bien : $D(C(x)) \equiv x^{cd} \equiv x \mod n$.

3 Pour chiffrer x, on a besoin de connaître c et n.
Le couple $(c\,;n)$ est appelé la `clé publique` car elle est connue de tous et répertoriée dans un annuaire.

4 Pour déchiffrer le message, il vaut connaître d et n.
d est appelé la `clé privée` car elle n'est connue que de la personne qui reçoit le message codé.

5 Les nombres p et q doivent rester confidentiels car leur connaissance entraîne celle de m, puis celle de d en résolvant l'équation de Bézout : $cd - km = 1$ (ce qui est possible car c est dans l'annuaire).

Le système RSA 1 024 bits correspond à un nombre n de l'ordre de $2^{1\,024}$, s'écrivant avec 309 chiffres.

Exemple

Pour simplifier les calculs, on prendra p et q petits, contrairement à la réalité.
On prend $p = 5$ et $q = 11$, donc $n = 55$.
Alors, $m = (5-1)(11-1) = 40$. Il nous faut trouver c et d tels que $cd \equiv 1 \mod 40$, $\text{pgcd}\,(c\,;\,m) = 1$.
Le premier nombre c possible est $c = 3$ (car 3 et 40 sont premiers entre eux). À l'aide d'un tableur, on calcule tous les nombres $3d$ possibles et on prend le premier qui vérifie $3d \equiv 1 \mod 40$: on trouve $d = 27$.
Les lettres de l'alphabet sont représentées par des nombres selon la règle suivante :

$$A \to 1 \quad ; \quad B \to 2 \quad ; \quad C \to 3 \quad ; \quad \ldots \quad ; \quad Z \to 26$$

Nous allons coder le mot : « MOT ».

Lettre	Nombre correspondant	Congruence	Nombre final
M	13	$13^3 \equiv 52 \mod 55$	52
O	15	$15^3 \equiv 20 \mod 55$	20
T	20	$20^3 \equiv 25 \mod 55$	25

Bien entendu, nous allons nous servir d'un tableur (par exemple) pour automatiser le cryptage :

	A	B	C	D	E	F	G	H
1	p	5		c	3		Vérif.	OK
2	q	11		d	27			
3								
4	Lettres :	M	O	T				
5		13	15	20				
6	Code :	52	20	25				

- Dans la cellule `H1` :
  On entre la formule : `=SI(MOD(E1*E2;(B1-1)*(B2-1))=1;"OK";"KO")`.
  Cela permet de vérifier que les valeurs de c et d conviennent.
- Dans la cellule `B5` :
  On entre la formule : `=CODE(B4)-64`.
  Ensuite, on copie la formule sur toute la ligne 5.
- Dans la cellule `B6` :
  On entre la formule : `=MOD(PUISSANCE(B5;$E$1);$B$1*$B$2)`.
  Ensuite, on la copie sur toute la ligne 6.

Pour déchiffrer « 52 », il faudrait théoriquement mettre dans la cellule `B7` la formule :

```
=MOD(PUISSANCE(B6;$E$2);$B$1*$B$2)
```

puis dans la cellule `B8` la formule :

```
=CAR(B7+64)
```

Cependant, 52^{27} étant trop grand pour la tableur, cela donnerait un résultat erroné.
Un tableur est donc approprié lorsque c et d ne sont pas trop grands. Sinon, il faut contourner le problème à l'aide d'un programme que l'on fera tourner en un langage quelconque.

Algorithme 1: Algorithme de cryptage RSA

```
Entrées
  n,c : entiers (pour la clé RAS)
  i,j : entiers (pour les boucles)
  r : entier
  LettreCodee, msg, msgCrypte : chaînes alphanumériques
Traitement
  Pour i allant de 1 à longueur(msg)
    r ← 1
    Pour j allant de 1 à c
      LettreCodee ← ascii(msg[j])-64
      r ← mod(r×LettreCodee;n)
    Fin du Pour
    msgCrypte[i] ← r
  Fin du Pour
Sortie
  Afficher msgCrypte
```

Algorithme 2: Algorithme de décryptage RSA

```
Entrées
  n,d : entiers (pour la clé RAS)
  i,j : entiers (pour les boucles)
  r : entier
  msg, msgDecrypte : chaînes alphanumériques
Traitement
  Pour i allant de 1 à longueur(msg)
    r ← 1
    Pour j allant de 1 à d
      r ← mod(r×msg[j];n)
    Fin du Pour
    msgDecrypte[i] ← lettreAscii(r+64)
  Fin du Pour
Sortie
  Afficher msgDecrypte
```

Complément 6: Problèmes de codage : calcul de clés

Que ce soit dans un R.I.B. (Relevé d'Identité Bancaire), dans un numéro de sécurité sociale, dans les codes barres ou les codes ISBN (International Standard Book Number), il y a toujours un dernier nombre qui sert de « clé de contrôle » et qui permet de vérifier que le codage n'est pas erroné.

1 Dans un R.I.B.

Un R.I.B. est constitué de la manière suivante :

Code Banque	Code guichet	N° compte	Clé
12345	25896	35715942681	?

La clé de contrôle est le reste de la division euclidienne de $N = 100a$ par 97, où a est le nombre constitué est 21 chiffres précédents.
Ici, $N = 452\,588\,963\,571\,594\,268\,100$. Nous pourrions effectuer la division euclidienne de N par 97 à la main, mais on peut aussi utiliser les congruences en remarquant que :

$$10^0 \equiv 1 \mod 97 \qquad 10^1 \equiv 10 \mod 97 \qquad 10^2 \equiv 3 \mod 97$$
$$10^3 \equiv 10^1 \times 10^2 \equiv 30 \mod 97 \qquad 10^4 \equiv 9 \mod 97 \qquad 10^6 \equiv 27 \mod 97$$

On pourrait raisonner de la même façon pour les puissances suivantes de 10. On a alors, par exemple :

$$\begin{aligned} N &= 1\,234 \times 10^{19} + 525\,896 \times 10^{13} + 357\,159\,426 \times 10^4 + 81 \times 10^2 \\ N &\equiv 1\,234 \times 17 + 525\,896 \times 15 + 357\,159\,426 \times 9 + 81 \times 3 \mod 97 \\ N &\equiv 3\,222\,344\,495 \mod 97 \\ N &\equiv 33 \mod 97 \end{aligned}$$

Et comme $97 - 33 = 64$, la clé du R.I.B. est donc 64.

2 Dans un code ISBN-13 et code barre EAN-13.

Un ISBN-13 ou un code barre EAN-13 est constitué de 13 chiffres suivis d'une clé.
Considérons le code ISBN suivant : 978-2-10-054640-x, où « x » est la clé de contrôle.
Pour calculer cette clé, on commence par calculer :

$$\begin{aligned} N &= (9+8+1+0+4+4) + 3 \times (7+2+0+5+6+0) \\ &= 26 + 3 \times 20 \\ &= 26 + 60 \\ &= 86 \\ &\equiv 6 \mod 10 \end{aligned}$$

Ensuite, on calcule :

$$\begin{aligned} &10 - 6 \mod 10 \\ \equiv &4 \mod 10 \end{aligned}$$

On obtient ainsi la clé : 4.

3 Dans un numéro de sécurité sociale.

Considérons le numéro suivant : 1 73 07 33 063 033 [clé].
On considère le nombre $N = 1\,730\,733\,063\,033$. Alors, à l'aide d'un tableur, en tapant dans une cellule la formule `=MOD(1730733063033;97)`, on trouve :

$$N \equiv 24 \mod 97.$$

De plus, $97 - 24 = 73$. Donc, la clé est égale à 97.

4 Sur un billet de banque en euros.

Sur les billets de banque (en euros), il y a une série de 12 chiffres précédés d'une lettre. Il existe un moyen rudimentaire de vérifier que ce « code » est valable.
Considérons un billet où est écrit : X27385267637.
Dans un premier temps, on remplace la lettre par son rang dans l'alphabet (1 pour A, 2 pour B, etc.) puis on forme un nombre N en remplaçant la lettre par ce rang. Ici, on aurait $N = 2\,427\,385\,267\,637$.
Ensuite, on calcule le reste de la division euclidienne de N par 9. Ici, il vaut 8.
Si ce reste n'est pas égal à 8, alors le billet est faux.

Complément 7: Critères de divisibilité.

1 Critère de divisibilité par 11.

Considérons un nombre $N = \overline{abcd}$ écrit en base 10. Donc, $N = a\times10^4+b\times10^3+c\times10^1+d\times10^0$. Or, $1\,000 \equiv -1 \mod 11$, $100 \equiv 1 \mod 11$, $10 \equiv -1 \mod 11$ et $1 \equiv 1 \mod 11$ donc :

$$N \equiv -a+b-c+d \mod 11.$$

Ainsi, $11|N \Leftrightarrow 11|(-a+b-c+d)$.
Par exemple, si $N = 9\,845$, alors $-a+b-c+d = -(9+4)+(8+5) = -13+13 = 0$ donc N est divisible par 11.
N.B. Pour un nombre plus grand, le principe est le même. Si $N = 98\,536\,295$, on calcule $-(9+5+6+9)+(8+3+2+5) = -29+18 = -11$, divisible par 11. Donc N est divisible par 11.

2 Critère de divisibilité par 7.

Considérons un nombre à au moins 3 chiffres. Alors, on peut l'écrire $N = 10p+u$.
Considérons alors le nombre $N' = p-2u$. Alors, $N-10N' = 10p+u-10(p-2u) = 21u$. Donc $N-10N'$ est divisible par 7 :

$$\begin{aligned} & N-10N' \equiv 0 \mod 7 \\ \Leftrightarrow & N \equiv 10N' \mod 7 \\ \Leftrightarrow & N \equiv N' \mod 7 \text{ car } \operatorname{pgcd}(10\,;\,7) = 1. \end{aligned}$$

Et donc :

$$N' \equiv 0 \mod 7 \Leftrightarrow N \equiv 0 \mod 7.$$

Prenons par exemple $N = 3\,983$. Alors, $N' = 398-2\times3 = 392$. En répétant l'algorithme sur N', on a : $N'' = 39-2\times2 = 35$. Or, $7|N''$ donc N est divisible par 7.

3 Critère de divisibilité par p, $p \in \mathbb{P}$.

On peut s'inspirer du point précédent pour trouver un critère de divisibilité par $p \in \mathbb{P}$. Considérons $N = 10p + u$ et $N' = p - ku$, $k \in \mathbb{N}^*$.
Alors, $N - 10N' = (10k + 1)u$. Comme $\text{pgcd}\,(10\,;\,10k + 1) = 1$, si $10k + 1 \equiv 0 \mod p$, alors $N \equiv 0 \mod p$.
On trouve à l'aide d'un tableur (par exemple) :

p	k
7	2
11	1
13	9
17	5
19	17
23	16
29	26
31	3

p	k
37	11
41	4
43	30
47	14
53	37
59	53
61	6
67	20

p	k
71	7
73	51
79	71
83	58
89	80
97	29

À part pour 7, 11, 17, 31 et 41 (éventuellement pour 61 et 71), les autres valeurs de k ne nous servent pas réellement pour vérifier mentalement si un nombre est divisible par p. Cependant, ces valeurs peuvent servir dans des programmes informatiques.

N.B. Il existe plus d'un critère de divisibilité pour un certain nombre p. Il n'y a qu'à regarder ce que nous avons trouvé pour 11 : nous en avons déjà trouvé 2 aux points 2 et 3. Le dernier critère trouvé pour 11 étant, pour $N = 9\,845$: $984 - 5 = 979 \to 97 - 9 = 88$ et $11|88$ donc $11|9\,845$.

Complément 8: Nombres en base a

Dans la vie de tous les jours, nous comptons en base 10 : cela signifie que tout nombre N se décompose selon l'écriture :

$$N = \sum_{i=0}^{n} 10^n x_i$$

On utilise la notation suivante :

$$N = 459\,015_{(10)} \qquad \text{ou} \qquad N = 45\,905_{10}$$

pour désigner le nombre en base 10.
Mais tout nombre peut s'exprimer en base autre que décimale. Par exemple, en informatique, on utilise la base 2 (le binaire). En base a, N s'exprime de la façon suivante :

$$N = \sum_{i=0}^{n} a^n y_i \ .$$

Les y_i s'obtiennent en faisant des divisions euclidiennes par a.
Prenons par exemple $N = 45\,905_{(10)}$. On a alors :

$$\begin{aligned}
45\,905 &= 2 \times 22\,952 + 1 \\
22\,952 &= 2 \times 11\,476 + 0 \\
11\,476 &= 2 \times 5\,738 + 0 \\
5\,738 &= 2 \times 2\,869 + 0 \\
2\,869 &= 2 \times 1\,434 + 1 \\
1\,434 &= 2 \times 717 + 0 \\
717 &= 2 \times 358 + 1 \\
358 &= 2 \times 179 + 0 \\
179 &= 2 \times 89 + 1 \\
89 &= 2 \times 44 + 1 \\
44 &= 2 \times 22 + 0 \\
22 &= 2 \times 11 + 0 \\
11 &= 2 \times 5 + 1 \\
5 &= 2 \times 2 + 1 \\
2 &= 2 \times 1 + 0 \\
1 &= 2 \times 0 + 1
\end{aligned}$$

On a alors :

$$45\,905_{(10)} = 1011001101010001_{(2)}$$

Algorithme 3: Algorithme de conversion en base a

Entrées

N : entier (en base 10)
M : chaîne numérique (contenant la conversion en base a)
a : base dans laquelle on veut convertir a
u : liste
n : entier de boucle

Traitement

u[0] ← N
n ← 1
M ← MOD(N;a)
Tant que n<ln(N)/ln(a)
u[n] ← (u[n-1]-y[n-1])/a
M ← MOD(u[n];a)+"M"
n ← n+1
Fin du Tant que

Sortie

Afficher M

```php
<?php // Permet de convertir n'importe quel nombre décimal en base a.
        if (!isset($_POST['N']) && !isset($_POST['a']))
        { // quand les valeurs de N et a ne sont pas données ...
        echo "<form action='".basename( __FILE__ )."' method='post'>";
        echo "N=<input type='text' size='10' name='N'> (nombre à convertir)<br>";
        echo "a=<input type='text' size='10' name='a'> (base en laquelle N
        sera converti)<br>";
        echo "<input type='submit' value='valider'>";
        echo "</form>";
        }
        else
        { // si N et a sont donnés ...
        $N = $_POST['N'];
        $a = $_POST['a'];
        $n = 1;
        $u[0]=$N;
        $M=$N%$a;
        while ($n<log($N,$a))
        { // n est nécessairement inférieur au log en base a de N
                $u[$n] = ($u[$n-1]-$y[$n-1])/$a;
                $M = $u[$n]%$a.$M;
                $n++;
        }
        echo $M;
        }
?>
```

Complément 9: Chiffrement de Hill (Pré-requis : calcul matriciel)

On considère un mot constitué de lettres : $L_1 L_2 \cdots L_n$, n étant pair.
Soient 4 nombres a, b, c et d qui vont constituer la clé du chiffrement.
On remplace L_i par sa position dans l'alphabet en convenant d'avoir 0 pour A, 1 pour B, etc. On obtient alors une suite de nombres : $u_1,\ u_2,\ \cdots,\ u_n$. On pose alors :

$$\forall k \in \left[\!\left[1\,;\frac{n}{2}\right]\!\right],\ \begin{pmatrix} c_{2k-1} \\ c_{2k} \end{pmatrix} \equiv \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} u_{2k-1} \\ u_{2k} \end{pmatrix} \mod 26$$

On obtient une suite de nombres $c_1,\ c_2,\ \cdots,\ c_n$ qui correspond à une suite de lettres.

Prenons par exemple le mot : CRYPTOGRAPHE et choisissons la clé $a = 3$, $b = 7$, $c = 2$ et $d = 13$. On a alors :

Lettres L_i	C	R	Y	P	T	O	G	R	A	P	H	E
Rang u_i	2	17	24	15	19	14	6	17	0	15	7	4

On a alors :

$$\begin{aligned} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} &= \begin{pmatrix} 3 & 7 \\ 2 & 13 \end{pmatrix} \begin{pmatrix} 2 \\ 17 \end{pmatrix} \\ &= \begin{pmatrix} 125 \\ 225 \end{pmatrix} \\ &\equiv \begin{pmatrix} 21 \\ 17 \end{pmatrix} \mod 26 \\ &\rightsquigarrow \begin{pmatrix} V \\ R \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \begin{pmatrix} c_3 \\ c_4 \end{pmatrix} &= \begin{pmatrix} 3 & 7 \\ 2 & 13 \end{pmatrix} \begin{pmatrix} 24 \\ 15 \end{pmatrix} \\ &= \begin{pmatrix} 177 \\ 243 \end{pmatrix} \\ &\equiv \begin{pmatrix} 21 \\ 9 \end{pmatrix} \mod 26 \\ &\rightsquigarrow \begin{pmatrix} V \\ J \end{pmatrix} \end{aligned}$$

Écrivons un algorithme pour éviter de tout faire à la main :

Algorithme 4: Algorithme de chiffrement de Hill

Entrées

```
a, b, c, d : clé du chiffrement
L : liste contenant les lettres du message à crypter
u : liste de nombres correspondant à la position des lettres
```

Traitement

Pour i allant de 0 à NbLettres(L)-1 avec un pas de 2

```
u[i] ← position(L[i])
u[i+1] ← position(L[i+1])
c[i] ← MOD(a×u[i]+b×u[i+1];26)
c[i+1] ← MOD(c×u[i]+d×u[i+1];26)
Afficher Lettre(c[i])Lettre(c[i+1])
```

Fin du Pour

En PHP, cela donne :

```php
<?php
if (!isset($_POST['message']) && !isset($_POST['a']) &&
 !isset($_POST['b'])&& !isset($_POST['c'])&& !isset($_POST['d']))
{
        echo "<form action='".basename( __FILE__ )."' method='post'>";
        echo "Message : <input type='text' size='50' name='message'><br>";
        echo "a=<input type='text' size='4' name='a'> 
        b=<input type='text' size='4' name='b'> 
        c=<input type='text' size='4' name='c'> 
        d=<input type='text' size='4' name='d'><br>";
        echo "<input type='submit' value='valider'>";
        echo "<form>";
}
else
{
        if ( strlen($_POST['message'])%2 ==0 )
        {
                $msg = strtoupper($_POST['message']);
                for ($i=0 ; $i<strlen($msg) ; $i=$i+2)
                {
                        // On commence par convertir la lettre en son code numérique
                        $u[$i] = ord($msg[$i])-65;
                        $u[$i+1] = ord($msg[$i+1])-65;
                        // On calcule maintenant le code
                        $c[$i] = ($_POST['a']*$u[$i]+$_POST['b']*$u[$i+1])%26;
                        $c[$i+1] = ($_POST['c']*$u[$i]+$_POST['d']*$u[$i+1])%26;
                        // On affiche les lettres correspondantes
                        echo chr($c[$i]+65).chr($c[$i+1]+65);
                }
        }
        else
                echo "Il faut un nombre pair de lettres.";
}
?>
```

Cela nous donne, pour notre exemple, le code : VRVJZMHZBNXO.

N.B. Dans le programme PHP, j'ai ajouté un test afin de vérifier si le nombre de lettre du message à coder est un nombre pair.

Le déchiffrement consiste à trouver les u_k avec la relation :

$$\forall k \in \left[\!\left[1\,;\frac{n}{2}\right]\!\right],\ \begin{pmatrix} u_{2k-1} \\ u_{2k} \end{pmatrix} \equiv \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} \begin{pmatrix} c_{2k-1} \\ c_{2k} \end{pmatrix} \mod 26$$

À l'aide d'un logiciel, on trouve que $\begin{pmatrix} 3 & 7 \\ 2 & 13 \end{pmatrix}^{-1} = \frac{1}{25}\begin{pmatrix} 13 & -7 \\ -2 & 3 \end{pmatrix}$.

Il nous faut trouver la valeur de $\frac{1}{25}$ modulo 26, c'est-à-dire le nombre x tel que $25 \times x \equiv 1 \mod 26$.

Or, $25 \equiv -1 \mod 26$ donc $25 \times 25 \equiv 1 \mod 26$. Ainsi, l'inverse de 25 modulo 26 est égal à 25 (ou -1 si on veut ... ce qui va nous arranger pour le calcul suivant).
On a alors :

$$\begin{aligned}
\begin{pmatrix} 3 & 7 \\ 2 & 13 \end{pmatrix}^{-1} &\equiv 25 \begin{pmatrix} 13 & -7 \\ -2 & 3 \end{pmatrix} \mod 26 \\
&\equiv -1 \begin{pmatrix} 13 & -7 \\ -2 & 3 \end{pmatrix} \mod 26 \\
&\equiv \begin{pmatrix} -13 & 7 \\ 2 & -3 \end{pmatrix} \mod 26 \\
&\equiv \begin{pmatrix} 13 & 7 \\ 2 & 23 \end{pmatrix} \mod 26
\end{aligned}$$

En faisant tourner le programme précédent avec le mot crypté comme premier mot, puis en prenant $a = 13$, $b = 7$, $c = 2$ et $d = 23$, on arrive au mot décrypté.

Complément 10: Chiffrement affine

On attribue à chaque lettre de l'alphabet un nombre (sa position) en convenant d'avoir 0 pour A, 1 pour B, etc.
On choisit deux nombres a et b. On note x le rang d'une lettre et $r(x)$ tel que $ax + b \equiv r(x) \mod 26$.
La lettre correspondant à $r(x)$ est la lettre codée.

- **Condition pour que ce chiffrement soit valable (pour que l'on puisse décoder tout message).**

  → Si $\text{pgcd}\,(a\,;\,26) \neq 1$, notons x et x' deux rangs différents. alors,

  $$\begin{aligned} r(x) - r(x') &= 26(q - q') + a(x - x') \\ &= 26(q - q') + a\frac{26}{\text{pgcd}\,(a\,;\,26)} \quad \text{en prenant } x \text{ et } x' \text{ tels que } x - x' = \frac{26}{\text{pgcd}\,(a\,;\,26)} \\ &= 26\left(q - q' + \frac{a}{\text{pgcd}\,(a\,;\,26)}\right) \\ &\equiv 0 \mod 26 \end{aligned}$$

  Il est donc possible de trouver deux x et x' différents tels que $r(x) \equiv r(x') \mod 26$.
  Dans ce cas, le décryptage est impossible.

  → Si $\text{pgcd}\,(a\,;\,26) = 1$, soient deux nombres x et x' tels que $r(x) = r(x')$. Alors,

  $$a(x - x') = 26(q - q').$$

  Or, $\text{pgcd}\,(a\,;\,26) = 1$, donc d'après le théorème de Gauss, 26 divise $(x - x')$. Ainsi, $x = x'$ car $-25 \leqslant r(x) - r(x') \leqslant 25$.
  Dans ce cas, on est assuré que le décryptage est possible (à une lettre codée ne correspond qu'une solution décodée).

  Il est donc nécessaire que a et 26 soient premiers entre eux.

  Exemple : On prend $a = 17$ et $b = 2$ puis on cherche à coder la lettre K. Son rang dans l'alphabet vaut 10 et $17 \times 10 + 2 \equiv 16 \mod 26$. La lettre de rang 16 est Q donc la lettre codée est Q.

- **Comment décoder une lettre ?**
  On part du principe que $y \equiv ax + b \mod 26$ est le rang de la lettre codée. Alors, en convenant de noter a' l'inverse de a modulo 26, on a :

  $$\begin{aligned} ax + b \equiv y \mod 26 &\Leftrightarrow aa'x + ba' \equiv a'y \mod 26 \\ &\Leftrightarrow x \equiv a'(y - b) \mod 26 \end{aligned}$$

  Exemple : pour décoder notre lettre Q, dont le rang est $y = 16$, on fait :

  $$\begin{aligned} x &\equiv 23(16 - 2) \mod 26 \\ &\equiv 322 \mod 26 \\ &\equiv 12 \times 26 + 10 \mod 26 \\ &\equiv 10 \mod 26 \end{aligned}$$

  On retrouve bien le rang de la lettre K.

Complément 11: Chiffrement de Vigénère

Dans un premier temps, on associe à chaque lettre de l'alphabet sa position (0 pour A, 1 pour B, etc.).
Prenons par exemple le mot MATHEMATIQUES à coder. Le chiffrement de Vigénère nécessite un mot clé ou une phrase clé. Prenons le mot « SYMPA ». Pour chiffrer notre message, on fera :

M	A	T	H	E	M	A	T	I	Q	U	E	S
12	0	19	7	4	12	0	19	8	16	20	4	18
S	Y	M	P	A	S	Y	M	P	A	S	Y	M
18	24	12	15	0	18	24	12	15	0	18	24	12
$12+18$ $=30$ $\equiv 4$	$0+24$ $=24$ $\equiv 24$	$19+12$ $=31$ $\equiv 5$	$7+15$ $=22$ $\equiv 22$	$4+0$ $=4$ $\equiv 4$	$12+18$ $=30$ $\equiv 4$	$0+24$ $=24$ $\equiv 24$	$19+12$ $=31$ $\equiv 5$	$8+15$ $=23$ $\equiv 23$	$16+0$ $=16$ $\equiv 16$	$20+18$ $=38$ $\equiv 12$	$4+24$ $=28$ $\equiv 2$	$18+12$ $=30$ $\equiv 4$
E	Y	F	W	E	E	Y	F	X	Q	M	C	E
$4-18$ $\equiv 12$	$24-24$ $\equiv 0$	$5-12$ $\equiv 19$	$22-15$ $\equiv 7$	$4-0$ $\equiv 4$	$4-18$ $\equiv 12$	$24-24$ $\equiv 0$	$5-12$ $\equiv 19$	$23-15$ $\equiv 8$	$16-0$ $\equiv 16$	$12-18$ $\equiv 20$	$2-24$ $\equiv 4$	$4-12$ $\equiv 18$
M	A	T	H	E	M	A	T	I	Q	U	E	S

Légende :

Message décodé
Clé
Message codé

Toutes les congruences sont modulo 26.

Algorithme 5: Algorithme de chiffrement de Vigénère

Entrées

```
message : liste (message à coder)
cle : liste (clé du chiffrement)
i : entier de boucle
```

Traitement

```
cle ← (⌊NbLettres(message)/NbLettres(cle)⌋+1)×cle
```

On forme ici une clé dont la longueur sera au moins égale à celle du message

Pour i allant de 0 à NbLettres(message)-1

```
Afficher Lettre(MOD(position(message[i])+position(cle[i]);26))
```

Fin du Pour

Voici un programme en PHP :

```php
<?php
 if (!isset($_POST['message']) && !isset($_POST['cle']))
 {
  echo "<form action='".basename( __FILE__ )."' method='post'>";
  echo "Phrase à chiffrer : <input type='text' size='100' name='message'><br>";
  echo "Clé : <input type='text' size='100' name='cle'><br>";
  echo "<input type='submit' value='valider'>";
```

```
8   echo "</form>";
9   }
10  else
11  {
12    $message = $_POST['message'];
13    $cle = str_repeat($_POST['cle'], ceil(strlen($message)/strlen($_POST['cle'])));
14    echo "Message codé : ";
15    for ($i=0 ; $i<strlen($message) ; $i++)
16    {
17      echo chr((ord($message[$i])-65+ord($cle[$i])-65)%26+65);
18    }
19  }
20 ?>
```