

# Algorithme d'Euclide étendu, Théorème de Bézout

La notion principale de cette partie du cours et celle du *plus grand commun diviseur* ou *pgcd* de deux entiers.

Le **plus grand commun diviseur** ou **pgcd** de deux entiers  $a$  et  $b$  non nuls est le plus grand entier qui divise à la fois  $a$  et  $b$ .

**Exemple**  $\text{pgcd}(12, 8) = 4$ .

Si les nombres sont suffisamment petits, il est possible de calculer leur pgcd à l'aide de la factorisation.

**Exemple** Calculer le pgcd de 84 et de 30.

- $84 = 2^2 \cdot 3 \cdot 7$
- $30 = 2 \cdot 3 \cdot 5$

On peut alors conclure que  $\text{pgcd}(84, 30) = 2 \cdot 3 = 6$ .

Cependant, lorsque les nombres sont grands, factoriser n'est pas toujours calculatoirement possible. Un algorithme plus efficace doit être utilisé. Un tel algorithme existe et est connu sous le nom de **algorithme d'Euclide**. Cet algorithme est basé sur une observation simple. Si  $a$  et  $b$  sont deux entiers tels que  $a > b$ , alors

$$\text{pgcd}(a, b) = \text{pgcd}(b, a \bmod b).$$

Cette observation peut être démontrée facilement. De manière générale, pour montrer l'égalité de deux entiers, il suffit de montrer qu'ils se divisent mutuellement.

- On montre que  $\text{pgcd}(a, b)$  divise  $\text{pgcd}(b, a \bmod b)$ .

On note  $d = \text{pgcd}(a, b)$ . Par la définition du symbole mod, il existe un entier  $k$  tel que  $a = kb + (a \bmod b)$ . Comme  $d$  est le plus grand commun diviseur de  $a$  et  $b$ ,  $d$  divise à la fois  $a$  et  $b$ . Il existe alors des entiers  $k_1$  et  $k_2$  tels que  $a = k_1 d$  et  $b = k_2 d$ . On a :

$$a \bmod b = a - kb = k_1 d - k(k_2 d) = (k_1 - k k_2) d,$$

et on conclue alors que  $d$  divise  $(a \bmod b)$ . Puisque  $d$  divise à la fois  $b$  et  $(a \bmod b)$  alors  $d$  divise  $\text{pgcd}(b, a \bmod b)$ .

Le sens réciproque se montre de façon similaire.

**Remarque** Puisque  $(a \bmod b) < a$ , on réduit le problème de trouver le pgcd de deux entiers donnés, à celui de trouver le pgcd de deux entiers plus petits.

## Description de l'algorithme

Voir [Algorithme d'Euclide](#)

**Exemple** Calculer le  $\text{pgcd}(243, 198)$ .

$$\begin{aligned} 243 &= 1 \cdot 198 + 45 \\ 198 &= 4 \cdot 45 + 18 \\ 45 &= 2 \cdot 18 + 9 \\ 18 &= 2 \cdot 9 + 0 \end{aligned}$$

Donc,

$$\begin{aligned} \text{pgcd}(243, 198) &= \text{pgcd}(198, 243 \bmod 198) \\ &= \text{pgcd}(198, 45) \\ &= \text{pgcd}(45, 198 \bmod 45) \\ &= \text{pgcd}(45, 18) \\ &= \text{pgcd}(18, 45 \bmod 18) \\ &= \text{pgcd}(18, 9) \\ &= \text{pgcd}(9, 18 \bmod 9) \\ &= \text{pgcd}(9, 0) \\ &= 9. \end{aligned}$$

## Théorème de Bézout

**Identité de Bézout** Soient  $a, b \in \mathbb{Z}$ . Il existe deux entiers  $u, v \in \mathbb{Z}$  tels que  $au + bv = \text{pgcd}(a, b)$ .

L'existence des entiers  $u$  et  $v$  est donnée par l'algorithme d'Euclide étendu (voir section suivante).

**Théorème de Bézout** Soient  $a, b \in \mathbb{Z}$ . Les entiers  $a, b$  sont premiers entre eux si et seulement s'il existe deux entiers  $u$  et  $v$  tels que  $au + bv = 1$ .

*Démonstration.* Si  $\text{pgcd}(a, b) = 1$ , par l'identité de Bézout, il existe deux entiers  $u$  et  $v$  tels que  $1 = ua + vb$ . Réciproquement, si on a une relation de la forme  $1 = ua + vb$ , alors un diviseur commun à  $a$  et à  $b$ , divise  $ua + vb$ , divise donc 1, et vaut alors  $\pm 1$ . On a bien montré que les entiers  $a$  et  $b$  sont premiers entre eux.

## Algorithme d'Euclide étendu

L'algorithme d'Euclide étendu est une variante de l'algorithme d'Euclide qui permet, à partir de deux entiers  $a$  et  $b$ , de calculer non seulement leur plus grand commun diviseur (pgcd), mais aussi un couple de *coefficients de Bézout*, c'est-à-dire deux entiers  $u$  et  $v$  tels que

$$au + bv = \text{pgcd}(a, b).$$

Cet algorithme est particulièrement utilisé lorsque on souhaite calculer l'inverse multiplicatif d'un entier.

La question importante est comment calcule-t-on les coefficients  $u$  et  $v$ . L'idée principale de l'algorithme est d'effectuer les mêmes étapes que pour l'algorithme d'Euclide, mais en exprimant à chaque itération le reste comme une combinaison linéaire de  $a$  et  $b$ . Puisque le dernier reste est le pgcd, celui-ci sera alors exprimé comme une combinaison linéaire de  $a$  et  $b$ .

### Déroulement de l'algorithme

On note  $r_0 = a$  et  $r_1 = b$ . On cherche  $u$  et  $v$  tels que  $\text{pgcd}(r_0, r_1) = u \cdot r_0 + v \cdot r_1$ .

$$r_0 = q_1 \cdot r_1 + r_2 \quad r_2 = u_2 \cdot r_0 + v_2 \cdot r_1$$

$$r_1 = q_2 \cdot r_2 + r_3 \quad r_3 = u_3 \cdot r_0 + v_3 \cdot r_1$$

$$\vdots \quad \vdots$$

$$r_{\ell-2} = q_{\ell-1} \cdot r_{\ell-1} + r_\ell \quad r_\ell = u_\ell \cdot r_0 + v_\ell \cdot r_1$$

De cette façon,  $\text{pgcd}(a, b) = r_\ell$ ,  $u = u_\ell$  et  $v = v_\ell$ .

**Exemple** On appliquera l'algorithme d'Euclide étendu pour  $a = r_0 = 243$  et  $b = r_1 = 198$ . Pour cela, on effectuera à gauche les étapes de l'algorithme d'Euclide (calcul des restes  $r_i$  et des quotients  $q_i$ .) En même temps on écrira à droite les calculs pour  $u_i$  et  $v_i$ , tels que  $r_i = u_i a + v_i b$ .

$$r_{i-2} = q_{i-1} \cdot r_{i-1} + r_i \quad r_i = [u_i]a + [v_i]b$$

$$243 = 1 \cdot 198 + 45 \quad 45 = [1]243 + [-1]198$$

$$\begin{aligned} 198 &= 4 \cdot 45 + 18 & 18 &= 198 - 4 \cdot 45 \\ & & &= 198 + (-4)(243 - 1 \cdot 198) \\ & & &= [-4]243 + [5]198 \end{aligned}$$

$$\begin{aligned} 45 &= 2 \cdot 18 + 9 & 9 &= 45 - 2 \cdot 18 \\ & & &= 45 - 2(198 - 4 \cdot 45) \\ & & &= 9 \cdot 45 - 2 \cdot 198 \\ & & &= 9 \cdot (243 - 198) - 2 \cdot 198 \\ & & &= [9]243 + [-11]198 \end{aligned}$$

$$18 = 2 \cdot 9 + 0$$

Nous avons alors  $\text{pgcd}(243, 198) = 9 = [9]243 + [-11]198$ .

On remarque en appliquant cette procédure que les combinaisons linéaires de la partie droite sont construites à l'aide des combinaisons linéaires *précédentes*. On va dériver maintenant des relations récursives pour calculer  $u_i$  et  $v_i$ .

### Relation récursive des coefficients de Bézout

Supposons qu'on se trouve à l'itération  $i$ . Pendant les deux itérations précédentes nous avons calculé:

$$r_{i-2} = [u_{i-2}]a + [v_{i-2}]b$$

$$r_{i-1} = [u_{i-1}]a + [v_{i-1}]b$$

Pendant l'itération  $i$  on calcule d'abord le quotient  $q_{i-1}$  et le nouveau reste  $r_i$  à partir de  $r_{i-1}$  et  $r_{i-2}$ :

$$r_{i-2} = q_{i-1} \cdot r_{i-1} + r_i.$$

Cette équation peut se réécrire ainsi:

$$r_i = r_{i-2} - q_{i-1} \cdot r_{i-1}.$$

Notre but est de représenter le nouveau reste  $r_i$  comme une combinaison linéaire de  $a = r_0$  et  $b = r_1$ . L'étape qui nous permet de faire cela est de substituer dans la dernière équation  $r_{i-2}$  de la première équation et  $r_{i-1}$  de la deuxième équation:

$$r_i = u_{i-2} \cdot a + v_{i-2} \cdot b - q_{i-1} \cdot (u_{i-1} \cdot a + v_{i-1} \cdot b).$$

En réarrangeant les termes on obtient:

$$\begin{aligned} r_i &= [u_{i-2} - q_{i-1} \cdot u_{i-1}]a + [v_{i-2} - q_{i-1} \cdot v_{i-1}]b \\ &= [u_i]a + [v_i]b. \end{aligned}$$

Ces formules récursives sont valables pour  $i \geq 2$ . On pose  $u_0 = 1, u_1 = 0$  et  $v_0 = 0, v_1 = 1$ .

## Calcul des inverses multiplicatifs modulo $n$

On peut utiliser l'algorithme d'Euclide étendu afin de calculer l'inverse modulaire d'un entier. Avant de continuer, on va démontrer un résultat crucial pour la démarche.

**Proposition** Un entier  $a$  est inversible modulo  $n$  si et seulement si  $\text{pgcd}(a, n) = 1$ .

*Démonstration* On suppose d'abord que l'entier  $a$  est inversible modulo  $n$ . Il existe alors  $b \in \mathbb{Z}/n\mathbb{Z}$  tel que  $ab \equiv 1 \pmod{n}$ . De cette congruence on voit alors qu'il existe un entier  $v$  tel que  $ab + nv = 1$ . Par le théorème de Bézout on peut alors conclure que  $\text{pgcd}(a, n) = 1$ .

Réciproquement, on suppose que  $\text{pgcd}(a, n) = 1$ . Par le théorème de Bézout, il existe des entiers  $u, v$  tels que  $au + nv = 1$ . Par conséquent,  $au \equiv 1 \pmod{n}$  et  $u$  est alors l'inverse de  $a$  modulo  $n$ .

Supposons maintenant qu'on veut calculer l'inverse de

$$a \pmod{n}, \text{ avec } n < a.$$

On vient de montrer que si cet inverse existe, alors forcément  $\text{pgcd}(a, n) = 1$ . En appliquant l'algorithme d'Euclide étendu on obtient un couple  $(u, v)$  tels que  $a \cdot u + n \cdot v = 1$ . On a alors

$$\begin{aligned} a \cdot u + n \cdot v &= 1 \\ a \cdot u + 0 &\equiv 1 \pmod{n} \\ a \cdot u &\equiv 1 \pmod{n}. \end{aligned}$$

La dernière équation est la définition de l'inverse. Ceci vaut dire que  $u$  est l'inverse de  $a$ :

$$u = a^{-1} \pmod{n}.$$

**Exercice** Calculer  $12^{-1} \pmod{67}$ .

**Solution** On applique l'algorithme d'Euclide étendu pour trouver le pgcd de 12 et 67.

$$\begin{aligned} 67 &= 5 \cdot 12 + 7 \\ 12 &= 1 \cdot 7 + 5 \\ 7 &= 1 \cdot 5 + 2 \\ 5 &= 2 \cdot 2 + 1 \\ 2 &= 2 \cdot 1 + 0 \end{aligned}$$

Le pgcd de 12 et 67 est alors bien 1.

On cherche les coefficients de Bézout.

$$\begin{aligned} 1 &= 5 - 2 \cdot 2 \\ &= 5 - 2(7 - 5) \\ &= -2 \cdot 7 + 3 \cdot 5 \\ &= -2 \cdot 7 + 3 \cdot (12 - 7) \\ &= 3 \cdot 12 - 5 \cdot 7 \\ &= 3 \cdot 12 - 5 \cdot (67 - 5 \cdot 12) \\ &= -5 \cdot 67 + 28 \cdot 12. \end{aligned}$$

Nous avons alors

$$\begin{aligned}-5 \cdot 67 + 28 \cdot 12 &= 1 \\ 28 \cdot 12 &\equiv 1 \pmod{67} \\ 28 &\equiv 12^{-1} \pmod{67}.\end{aligned}$$

## $\mathbb{Z}/n\mathbb{Z}$ est un corps $\Leftrightarrow n$ est premier

On va montrer maintenant que si  $p$  est un nombre premier alors  $\mathbb{Z}/p\mathbb{Z}$  est un corps.

**Proposition** Si  $p$  est un nombre premier, alors  $\mathbb{Z}/p\mathbb{Z}$  est un corps.

*Démonstration* On a déjà vu que  $\mathbb{Z}/p\mathbb{Z}$  est un anneau commutatif. Il suffit juste de montrer que tous les éléments non-nuls de  $\mathbb{Z}/p\mathbb{Z}$  sont inversibles. Les éléments non-nuls de  $\mathbb{Z}/p\mathbb{Z}$  sont les éléments  $1, 2, \dots, p-1$ . Puisque  $p$  est premier, tous les éléments non-nuls strictement inférieurs à  $p$  sont premiers avec  $p$ . Par conséquent, ils sont inversibles modulo  $p$ .

Montrer que si  $\mathbb{Z}/n\mathbb{Z}$  est un corps, alors  $n$  est un nombre premier est laissé comme exercice.



2011-2020 Mélanie Boudard <<http://melanie.boudard.free.fr/>>, Christina Boura <<http://christina-boura.info/en/content/home>>, Luca De Feo <<http://defeo.lu>>, licensed under the Creative Commons 4.0 Attribution-ShareAlike <<http://creativecommons.org/licenses/by-sa/4.0/>>.