Les algorithmes d'Euclide

N. B. Souvent, pour simplifier l'algorithme, certains documents n'écrivent pas le nom de la variable de sortie dans la déclaration de la fonction. C'est ce que nous faisons aussi ici.

1 L'algorithme d'Euclide initial (ou original)

Exercice 1. Soient a, b deux entiers non tous nuls avec $a > b$. Montrer que $\mathrm{pgcd}(a, b) = \mathrm{pgcd}(a - b, b)$. Autrement dit,

$$\mathrm{pgcd}(a, b) = \mathrm{pgcd}(\max(a, b) - \min(a, b), \min(a, b)).$$

Solution 1. Posons $d = \mathrm{pgcd}(a, b)$ et $c = \mathrm{pgcd}(a, a - b)$. Montrons que $d = c$. Il existe $k, l \in \mathbb{N}$ tels que $a = dl$ et $b = dl$, ce qui donne $a - b = dk - dl = d(k - l)$. Ainsi $d|a - b$. Donc d est un diviseur commun à a et $a - b$. Alors

$$d \leqslant c. \tag{1}$$

De même, il existe $a, x \in \mathbb{N}$ tels que $a = cx$ et $a - b = cy$, ce qui donne

$$b = a - cy = cx - cy = c(x - y).$$

Ainsi, $c|b$. Donc c est un diviseur commun à a et b. D'où

$$c \leqslant d. \tag{2}$$

D'après les deux inégalités (1) et (2), on a $d = c$, i.e. $\mathrm{pgcd}(a, b) = \mathrm{pgcd}(b, a - b)$.

Exercice 2. Soient $a, b \in \mathbb{N}$ avec $a \geqslant b > 0$. Montrer que

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

Solution 2. Soit $d = \mathrm{pgcd}(a, b)$ et $e = \mathrm{pgcd}(b, a \bmod b)$. Il existe $k, l \in \mathbb{N}$ tels que $a = dl$ et $b = dl$. En effectuant la division euclidienne de a par b, il existe $q, r \in \mathbb{N}$ tels que $a = bq + r$ avec $0 \leqslant r < b$. Puisque

$$r = a - bq = dk - dlq = d(k - lq),$$

l'entier d divise également $a \bmod b = r$. Ainsi, d divise à la fois b et $a \bmod b$, ce qui implique que

$$d \leqslant e. \tag{3}$$

Comme e divise b et $r = a \bmod b$, il existe $x, y \in \mathbb{N}$ tels que $b = ex$ et $r = ey$. Or

$$a = bq + r = exq + ey = e(xq + y).$$

Ainsi, e est un diviseur commun à a et b, ce qui implique que

$$e \leqslant d. \tag{4}$$

D'après les deux inégalités (3) et (4), on a $d = e$, i.e. $\mathrm{pgcd}(a, b) = \mathrm{pgcd}(b, a \bmod b)$.

Exercice 3. Montrer que l'algorithme d'Euclide original suivant trouve bien $\mathrm{pgcd}(a, b)$.

Algorithme d'Euclide original

```
Entrée = Deux entiers a et b
Sortie = Le PGCD de a et b

fonction euclide(a, b)
    tant que a ≠ b
        si a > b alors
            a := a - b
        sinon
            b := b - a
    renvoyer a
```

Solution 3. À chaque itération de l'algorithme initial d'Euclide, a et b sont mis à jour comme suit :

$$a_{i+1} = \max(a_i, b_i) - \min(a_i, b_i) \tag{5}$$

$$b_{i+1} = \min(a_i, b_i) \tag{6}$$

avec $a_0 = a$ et $b_0 = b$ et $i = 0, 1, 2, \ldots$.
D'après (6), il est clair que $b_{i+1} \leqslant b_i$, i.e. la suite (b_i) est décroissante. D'après le théorème du bon ordre, elle est stationnaire à partir d'une certaine indice N, i.e. $a_N = b_N$. Alors, d'après l'exercice 1,

$$\mathrm{pgcd}(a, b) = \mathrm{pgcd}(a_i, b_i) = \cdots = \mathrm{pgcd}(a_N, b_N) = a_N.$$

2 L'algorithme d'Euclide récursif

Exercice 4. Montrer que l'algorithme d'Euclide récursif suivant trouve bien $\mathrm{pgcd}(a, b)$.

Algorithme d'Euclide récursif

```
Entrée = Deux entiers a et b
Sortie = Le PGCD de a et b

fonction euclide(a, b)
    si b = 0 alors renvoyer a
    sinon renvoyer euclide(b, a modulo b)
```

Solution 4. A chaque étape de l'algorithme récursif calcule

$$\begin{aligned} \text{pgcd}(b, r_1) \text{ avec } r_1 = a \bmod b \text{ et } 0 \leqslant r_1 < b, \\ \text{pgcd}(r_1, r_2) \text{ avec } r_2 = b \bmod r_1, \text{ et } 0 \leqslant r_2 < r_1, \\ \vdots \\ \text{pgcd}(r_i, r_{i-1}) \text{ avec } r_i = r_{i-1} \bmod r_{i-2}, \\ \vdots, \end{aligned}$$

qui sont tous égaux à $\text{pgcd}(a, b)$ d'après l'Exercice 2. Puisque la suite (r_i) est strictement décroissante, et la relation $\leqslant$ une relation de bon ordre, la suite (r_i) est finie et constante à partir d'un certain rang N. Dans ce cas, on a vu en cours que r_N est la valeur la plus petite de cette suite ; nous allons montrer que $r_N = 0$. Si $r_N \neq 0$, on peut diviser r_{N-1} par r_N pour obtenir

$$r_{N+1} = r_{N-1} \bmod r_N \text{ avec } 0 \leqslant r_{N+1} < r_N,$$

ce qui contredit le fait que r_N est minimal. Donc forcément $r_N = 0$. L'algorithme s'arrête et trouve alors, d'après la condition initiale,

$$\text{pgcd}(r_{N-1}, r_N) = \text{pgcd}(r_{N-1}, 0) = r_{N-1} = \text{pgcd}(a, b).$$

3 L'algorithme d'Euclide itératif

Exercice 5. Montrer que l'algorithme d'euclide récursif suivant trouve bien $\text{pgcd}(a, b)$:

```
Algorithme d'Euclide itératif

fonction itpgcd(a, b)
r[-2] := a
r[-1] := b
i:=0
tant que r[i-1] n'est pas égal à 0, faire
       r[i] := r[i-2] mod r[i-1]
       i:=i+1
retourner(r_[i-2])
fin
```

Solution 5. Cet algorithme calcule les restes $r_1, r_2, \ldots$ et d'après l'Exercice 4, le pgcd de a et b est le dernier reste non nul, ce que l'algorithme trouve.

4 L'algorithme d'Euclide étendu

Exercice 6. Cet algorithme ajoute les variables et équations supplémentaires pour trouver les coefficients de Bézout s, t tels que

$$\text{pgcd}(a, b) = sa + tb.$$

Dans cet algorithme, on a besoin du quotient q_i de r_{i-2} par r_{i-1} :

$$q_i = r_{i-2} \text{ div } r_{i-1}.$$

On pose

$$\begin{cases} s_i &= s_{i-2} - q_i r_{i-1}, \\ t_i &= t_{i-2} - q_i t_{i-1} \end{cases}$$

avec les conditions initiales

$$\begin{cases} s_{-2} = 1, s_{-1} = 0 \\ t_{-2} = 0, s_{-1} = 1. \end{cases}$$

Avec les notations de l'Exercice 6, pour l'entier N tel que $r_N = 0$, on a

$$\text{pgcd}(a, b) = s_{N-1}a + t_{N-1}b. \tag{7}$$

Nous avons alors l'algorithme suivant :

```
Algorithme d'Euclide étendu

fonction etpgcd(a, b)
r[-2] := a
r[-1] := b
s[-2] := 1
s[-1] := 0
t[-2] := 0
t[-1] := 1
 i:=0
tant que r[i-1] n'est pas égal à 0, faire
       q[i] := r[i-2] div r[i-1]
       r[i] := r[i-2] mod r[i-1]
       s[i] := s[i-2]-q[i]r[i-1]
       t[i] := t_[i-2]-q[i]t[i-1]
       i:=i+1
retourner(s[i-2],t[i-2], r_[i-2])
fin
```

Soit N tel que $r_{N-1} = \text{pgcd}(a, b)$ (dernier reste non nul).
(1) Montrer que $r_i = s_i a + t_i b$ pour $i = 1, \ldots, N-1$ (indication : par récurrence sur forte i).
(2) Montrer que cet algorithme donne bien les coefficients de Bézout, en prouvant l'équation (7).

Solution 6. (1) Pour montrer que $r_i = s_i \cdot a + t_i \cdot b$ à chaque itération i, nous pouvons procéder par récurrence sur i.
Cas de base : pour $i = 0$, nous avons

$$r_0 = s_0 \cdot a + t_0 \cdot b = a - (a \bmod b) \cdot b = a \mod b$$

Donc l'équation est vérifiée pour $i = 0$.
Hypothèse de récurrence forte (HR) : supposons que $r_i = s_i \cdot a + t_i \cdot b$ soit vraie jusqu'à une certaine itération i ($\geqslant 1$) .

Étape de récurrence : nous voulons montrer que l'équation est également vraie pour $i = i + 1$. Or :

$$\begin{aligned} r_{i+1} &= r_{i-1} - q_i \cdot r_i \\ &= r_{i-1} - q_i(s_i a + t_i b) \text{ (par HR)} \\ &= (s_{i-1}a + t_{i-1}b) - q_i s_i a - q_i t_i b \text{ (par HR)} \\ &= (s_{i-1} - q_i s_i)a + (t_{i-1} - q_i t_i)b \text{ (on factorise } a \text{ et } b \text{)} \\ &= s_{i+1}a + t_{i+1}b \text{ (par définition de } s_{i+1} \text{ et } t_{i+1}) \end{aligned}$$

L'équation est donc vraie pour $k + 1$. Par récurrence, la formule $r_i = s_i \cdot a + t_i \cdot b$ est vraie pour tout $i = 1, \ldots, N - 1$.
(2) Pour $i = N - 1$, nous avons

$$r_{N-1} = \text{pgcd}(a, b) = s_{N-1}a + t_{N-1}b.$$