Exercice 4 (Algorithme d'Euclide (suite)).

Pour tout couple d'entiers $a \in \mathbb{Z}$ et $b \in \mathbb{N}$, le théorème de Bézout établit l'existence d'un couple $(u, v) \in \mathbb{Z}^2$ d'entiers tel que $au + bv = \text{pgcd}(a, b)$.

L'algorithme de la Figure 1, appelé **algorithme d'Euclide étendu**, donne une méthode pratique pour calculer un tel couple.

Dans cet algorithme, les variables notées avec un indice s permettent de stocker en permanence l'état courant et l'état suivant (indice s) des trois variables `d`, `u` et `v`.

Généralement, pour comprendre le fonctionnement de cet algorithme, on utilise un tableau de suivi de variables. Voici par exemple le tableau de suivi de variables pour les valeurs d'entrée $a = 61$ et $b = 9$:

Itération	d	d_s	q	r	u	v	u_s	v_s
Init.	61	9	non défini	non défini	1	0	0	1
Fin itération 1	9	7	6	7	0	1	1	-6
Fin itération 2	7	2	1	2	1	-6	-1	7
Fin itération 3	2	1	3	1	-1	7	4	-27
Fin itération 4	1	$\boxed{0}$	2	0	$\boxed{4}$	$\boxed{-27}$	peu importe	peu importe
Fin de la boucle								

Sur cet exemple, on vérifie donc que l'algorithme fonctionne correctement car on trouve bien que la dernière valeur prise par `d` est 1, ce qui correspond bien au PGCD de 61 et de 9. De plus, on a bien

$$1 = 4 \times 61 + (-27) \times 9$$

$u = 4$ et $v = -27$ sont donc bien des entiers de Bézout valides dans ce cas.

1. Appliquer cet algorithme aux entiers $a = 63$ et $b = 11$ en remplissant un tableau de suivi de variables comme ci-dessous. Conclure pour cet exemple.

2. Écrire une fonction C

```
int euclide_etendu(int a, int b, int *u, int *v)
```

qui reçoit deux paramètres entiers `a` et `b` et qui calcule les entiers u, v, d. On respectera soigneusement le prototype imposé.

3. Justifier la *terminaison* de cette fonction.

4. Montrer que la propriété suivante est un *invariant de boucle.*

$$\mathcal{P} : \quad au + bv = d$$

Vous pouvez introduire toutes les notations qui vous semblent nécessaires.

On pourra commencer par le vérifier sur un exemple, en annotant **proprement** et de façon lisible le tableau de la première question.

5. En déduire la *correction* de votre fonction.

6. Démontrer le théorème de Bézout

Corrigé de l'exercice 4.

[Retour à l'énoncé]

1. Voici le tableau de suivi de variables obtenu pour les entrées $a = 63$ et $b = 11$:

Itération	d	d_s	q	r	u	v	u_s	v_s
Init.	63	11	non défini	non défini	1	0	0	1
Fin itération 1	11	8	5	8	0	1	1	-5
Fin itération 2	8	3	1	3	1	-5	-1	6
Fin itération 3	3	2	2	2	-1	6	3	-17
Fin itération 4	2	1	1	1	3	-17	-4	23
Fin itération 5	1	$\boxed{0}$	2	0	$\boxed{-4}$	$\boxed{23}$	peu importe	peu importe
Fin de la boucle								

Les coefficients de Bézout obtenus sont donc : $u = -4$ et $v = 23$. Ils fonctionnent effectivement car :

$$1 = 63 \times (-4) + 11 \times 23$$

2. Voici une implémentation en C de l'algorithme d'Euclide étendu :

```c
int euclide_etendu(int a, int b, int *u, int *v)
{
  int d; // valeur courante
  int ds, us, vs; // valeurs suivantes

  int r, q;
  int tmpu, tmpv; // buffer de
                  // sauvegarde temporaire

   d = a;  // d_0 = a
  *u = 1;
  *v = 0;

  ds = b; // d_1 = b
  us = 0;
  vs = 1;

  while (ds != 0) // (d_k) variant
  {

    // invariant de boucle
    r = d % ds;
    q = d / ds;
    printf("%4d %4d %4d %4d\n", ds, q, us, vs);
    tmpu = us;
    tmpv = vs;
    us = *u - q*us;
    vs = *v - q*vs;
    *u = tmpu;
    *v = tmpv;
    d = ds;
    ds = r;
  }

  return d;
}
```

3. La preuve de terminaison de cet algorithme est totalement identique à celle de premier exercice sur l'algorithme d'Euclide simple en prenant la suite $(d_k)_{k\in\mathbb{N}}$ comme invariant de boucle.

4. Vérifions sur notre tableau de suivi de variable que

$$\mathcal{P} : \quad au + bv = d$$

est bien un invariant de boucle

k	d	q	u	v	$\mathcal{P}$
0	63	non défini	1	0	$63 \times 1 + 11 \times 0 = 63$
1	11	5	0	1	$63 \times 0 + 11 \times 1 = 11$
2	8	1	1	-5	$63 \times 1 + 11 \times (-5) = 8$
3	3	2	-1	6	$63 \times (-1) + 11 \times 6 = 3$
4	2	1	3	-17	$63 \times 3 + 11 \times (-17) = 189 - 187 = 2$
5	$\boxed{1}$	2	$\boxed{-4}$	$\boxed{23}$	$63 \times (-4) + 11 \times 23 = -252 + 253 = 1$
6	0	fin			

Pour prouver la correction de l'algorithme, nous allons utiliser l' invariant de boucle :

$$\mathcal{P} : \quad au + bv = d$$

où a, b, u, v et d désignent les valeurs associées aux variables `a`, `b`, `u`, `v` et `d`.
On note u_s, v_s et d_s les valeurs associées aux variables `us`, `vs` et `ds` qui correspondent aux valeurs que prendront `u`, `v` et `d` au tour suivant.

Initialisation : Au tout début du tout premier tour de boucle $d = a$, $u = 1$ et $v = 0$.
On a bien :

$$au + bv = a \times 1 + b \times 0 = a = d\,,$$

et la propriété est vérifiée.
On a aussi $d_s = b$, $u_s = 0$ et $v_s = 1$ et

$$au_s + bv_s = a \times 0 + b \times 1 = b = d_s\,,$$

et la propriété est également vérifiée.

Conservation : On note u, v et d désignent les valeurs associées aux variables `u`, `v` et `d` au tout début d'un tour de boucle et u', v' et d' leurs valeurs à la fin d'un tour de boucle.
On suppose la propriété vraie au tout début d'un tour de boucle quelconque pour `u`, `v`, `d` et pour `us`, `vs`, `ds` :

$$au + bv = d$$

$$au_s + bv_s = d_s$$

On note u', v', d', u'_s, v'_s et d'_s les valeurs de ces variables en fin de tour. On note q et r le contenu des variables `q` et `r` qui ne varie pas entre le début et la fin d'un tour de boucle.
D'après l'algorithme :

$$\begin{aligned} d &= qd_s + r \\ u'_s &= u - qu_s \\ v'_s &= v - qv_s \\ u' &= u_s \\ v' &= v_s \\ d' &= d_s \\ d'_s &= r \end{aligned}$$

On a donc :

$$au' + bv' = au_s + bv_s = d_s = d'$$

et

$$au'_s + bv'_s = a(u - qu_s) + b(v - qv_s) = \underbrace{au + bv}_{=d} - q\,(\underbrace{au_s + bv_s}_{d_s}) = d - qd_s = r = d'_s$$

On a donc bien montré que la propriété était conservée en fin de tour.

5. L'algorithme d'Euclide étendu n'est qu'une amélioration de l'algorithme d'Euclide vu à l'exercice 2. On a donc déjà démontré que la valeur d' retournée à la fin de l'algorithme, à la sortie du dernier tour de boucle, correspondait bien au PGCD de a et de b :
Par ailleurs, à la sortie de dernier tour de boucle, on a aussi :

$$au' + bv' = d' = \mathrm{pgcd}(a, b)$$

ce qui montre que les valeurs u' et v' renvoyées en sortie de l'algorithme sont des coefficients de Bézout valides.

Ceci achève la démonstration de la correction de l'algorithme et donc, la terminaison ayant été prouvée en question3, la preuve de correction totale.

6. On va démontrer le théorème de Bézout :

$$\forall a \in \mathbb{Z},\ \forall b \in \mathbb{N},\ \exists (u_0, v_0) \in \mathbb{Z}^2,\ au_0 + bv_0 = \text{pgcd}(a, b)$$

Démonstration

Si $b = 0$, le théorème est trivial : il suffit de choisir $u = 1$ et v quelconque et on a toujours

$$a \times 1 + 0 \times v = \text{pgcd}(a, 0)$$

en se rappelant que $\text{pgcd}(a, 0) = a$.
On suppose donc dans la suite que $b > 0$

a. On définit l'ensemble :

$$E = \left\{ d \in \mathbb{N}^* \text{ tels que } d \text{ s'écrit sous la forme } au + bv,\ (u, v) \in \mathbb{Z}^2 \right\}$$

E est non vide car il contient $b = a \times 0 + b \times 1$ qui est positif ($b > 0$ par hypothèse).
$E \subset \mathbb{N}$ par définition, et parce que a, b, u et v sont des entiers.
E admet donc un plus petit élément d_0 :

$$\exists (u_0, v_0) \in \mathbb{Z}^2,\ au_0 + bv_0 = d_0$$

On a $d_0 > 0$ par définition de E.

b. Montrons que $\text{pgcd}(a, b) | d_0$. Par définition du PGCD, $\text{pgcd}(a, b) | a$ et $\text{pgcd}(a, b) | b$ donc $\text{pgcd}(a, b) | d_0$

c. Montrons que $d_0 | a$. Écrivons la division euclidienne de a par d_0 :

$$a = qd_0 + r,\ 0 \leq r < d_0$$

$$r = a - qd_0 = a - q(au_0 + bv_0) = a(1 - qu_0) + b \times (-v_0)$$

On a $r \geq 0$ (c'est un reste) donc il y a deux possibilités $r > 0$ ou $r = 0$:

Si $r > 0$: L'écriture obtenue montre alors que $r \in E$. Mais $r < d_0$ contredit la minimalité de d_0 et on obtient donc une contradiction ce qui rend ce cas impossible

On a donc $r = 0$: c'est-à-dire $a = qd_0$, donc $d_0 | a$.

d. De manière tout à fait analogue, on montre que $d_0 | b$.

e. Montrons que $d_0 | \text{pgcd}(a, b)$. On a montré que $d_0 | a$ et $d_0 | b$. Donc d_0 est un divisuer commun à a et à b. Par maximalité du PGCD, on a donc $d_0 | \text{pgcd}(a, b)$

f. Conclusion : On a montré que $\text{pgcd}(a, b) | d_0$ et $d_0 | \text{pgcd}(a, b)$ donc $d_0 = \text{pgcd}(a, b)$ ou $\text{pgcd}(a, b) = -d_0$ mais comme ces deux quantités sont par définition positives, on a en fait :

$$d_0 = \text{pgcd}(a, b)$$

On a donc montré qu'il existait $(u_0, v_0) \in \mathbb{Z}^2$ tels que :

$$au_0 + bv_0 = d_0 = \text{pgcd}(a, b)$$

Algorithme 1 : `euclide_etendu`

Donnée : a, `entier`
Donnée : b, `entier`

Variable de travail : d, `entier`
Variable de travail : u, `entier`
Variable de travail : v, `entier`

Variable de travail : tmp_u, `entier`
Variable de travail : tmp_v, `entier`

Variable de travail : d_s, `entier`
Variable de travail : u_s, `entier`
Variable de travail : v_s, `entier`

Variable de travail : r, `entier`
Variable de travail : q, `entier`

1 $d \leftarrow a$
2 $u \leftarrow 1$
3 $v \leftarrow 0$
4 $d_s \leftarrow b$
5 $u_s \leftarrow 0$
6 $v_s \leftarrow 1$
7 *tant que* $d_s \neq 0$ *faire*
8 $(q, r) \leftarrow$ (quotient, reste) de la division euclidienne de d par d_s
9 $tmp_u \leftarrow u_s$
10 $tmp_v \leftarrow v_s$
11 $u_s \leftarrow u - q \times u_s$
12 $v_s \leftarrow v - q \times v_s$
13 $u \leftarrow tmp_u$
14 $v \leftarrow tmp_v$
15 $d \leftarrow d_s$
16 $d_s \leftarrow r$
17 **Renvoyer** les valeurs d, u et v.

FIGURE 1 – Algorithme d'Euclide étendu écrit en pseudo-code.