

CI 2 : ALGORITHMIQUE & PROGRAMMATION

CHAPITRE 3 – VARIANT ET INVARIANT DE BOUCLE

Savoir

SAVOIRS :

- justifier qu’une itération (ou boucle) produit l’effet attendu au moyen d’un invariant ;
- démontrer qu’une boucle se termine effectivement.

L’algorithme suivant permet de réaliser une division euclidienne :

Pseudo Code

```
Data :  $a, b \in \mathbb{N}^*$ 
reste  $\leftarrow$  a
quotient  $\leftarrow$  0
tant que  $reste \geq b$  faire
|   reste  $\leftarrow$  reste – b
|   quotient  $\leftarrow$  quotient + 1
fin
```

Problématique

- Comment s’assurer que l’algorithme en question réalise bien une division euclidienne ?
- Comment s’assurer que la boucle while se termine ?

1	Variant	1
1.1	Définition	1
1.2	Exemple – L’algorithme d’Euclide.	2
2	Invariant de boucle	2
2.1	Problématique	2
2.2	Exemple division euclidienne	3
2.3	Exemple – L’algorithme d’Euclide.	3

Ce document évolue. Merci de signaler toutes erreurs ou coquilles.

1 Variant

1.1 Définition

En utilisant l’exemple précédent de la division euclidienne, pour sortir de la boucle, il est nécessaire que le reste devienne, au cours de l’algorithme, inférieur à la valeur de b . Dans ce cas, on constate que b étant un entier naturel, la valeur du reste va diminuer à chaque tour de boucle. Ainsi, on peut intuitivement que cet algorithme se terminera.

Définition

Variant de boucle

Soit une condition booléenne permettant de sortir d'une boucle constituée d'une comparaison entre une variable et une constante de types entiers positifs. La variable est un variant de boucle si :

- elle reste positive tout au long de l'algorithme ;
- elle décroît (ou croît) strictement à chaque itération de la boucle.

Ainsi, après un nombre fini d'itérations, on est sûr que la boucle se terminera.

Remarque

Un variant de boucle **permet** de s'assurer qu'une boucle se terminera.

Un variant de boucle **ne permet pas** de s'assurer qu'un algorithme fournit la réponse attendue.

1.2 Exemple – L'algorithme d'Euclide.

Cet algorithme permet de calculer le PGCD de deux nombres entiers. Il se base sur le fait que si a et b sont deux entiers naturels non nuls, $pgcd(a, b) = pgcd(b, a \bmod b)$.

Pseudo Code

```
Data :  $a, b \in \mathbb{N}^*, a < b$ 
 $x \leftarrow a$ 
 $y \leftarrow b$ 
tant que  $y \neq 0$  faire
     $r \leftarrow$  reste de la division euclidienne de  $x$ 
    par  $y$ 
     $x \leftarrow y$ 
     $y \leftarrow r$ 
fin
```

Exemple

Calculons le PGCD de 240 et 64 :

$$240 = 3 \cdot 64 + 48$$

$$64 = 1 \cdot 48 + 16$$

$$48 = 3 \cdot 16 + 0$$

Dans cet exemple, la condition pour sortir de la boucle est que y devienne nul. y est un variant de boucle si y est un entier positif qui décroît strictement à chaque itération de boucle. Dans ces conditions, on est sûr que la boucle se terminera.

Le reste de la division euclidienne de x par y , r , est un entier positif strictement inférieur à y . De plus, à chaque itération, y prend la valeur de r . y décroît donc à chaque itération.

En conséquence, y est un variant de boucle.

2 Invariant de boucle

2.1 Problématique

Lors de la réalisation d'un algorithme, il est nécessaire

- de vérifier que celui-ci permet bien de répondre au problème initial ;
- de s'assurer que celui-ci se termine sans quoi le résultat du problème ne sera jamais délivré à l'utilisateur.

On s'intéresse ici uniquement aux structures itératives. (Les structures récursives seront traitées en seconde année.)

Invariant de boucle [1]

Un invariant de boucle est une propriété ou une formule logique,

- qui est vérifiée après la phase d'initialisation ;
- qui reste vraie après l'exécution d'une itération ;
- et qui, conjointement à la condition d'arrêt, permet de montrer que le résultat attendu et bien le résultat calculé.

[2]

1. Définir les préconditions (état des variables avant d'entrer dans la boucle).
2. Définir un invariant de boucle.
3. Prouver l'invariant (correspond à $\mathcal{P}(n) \Leftarrow \mathcal{P}(n+1)$).
4. Montrer la terminaison du programme.
5. Condition de sortie de boucle + invariant de boucle \Leftarrow postcondition.

2.2 Exemple division euclidienne

Utilisons la méthode définie précédemment.

1. Initialement, l'utilisateur saisit les entiers a et b . Le reste prend la valeur de a et me quotient prend la valeur 0.
2. On postule que la quantité $b \cdot \text{quotient} + \text{reste} = a$ est un invariant de boucle.
3. Démontrons par récurrence le postulat précédent.
 - Avant d'entrer dans la boucle, on a bien $b \cdot \text{quotient}_0 + \text{reste}_0 = b \cdot 0 + a = a$. Le postulat est donc vrai.
 - Admettons le postulat vrai au rang n : $b \cdot \text{quotient}_n + \text{reste}_n = a$.
 - Montrons que le postulat est vrai au rang $n+1$:

$$b \cdot \text{quotient}_{n+1} + \text{reste}_{n+1} = b (\text{quotient}_n + 1) + \text{reste}_n - b = b \text{quotient}_n + \text{reste}_n + b - b = a$$

La relation est donc vraie au rang $n+1$.

4. On a montré la terminaison du programme grâce au variant de boucle.
5. En sortie de programme, l'invariant de boucle reste vérifié.

2.3 Exemple – L'algorithme d'Euclide.

Cet algorithme permet de calculer le PGCD de deux nombres entiers. Il se base sur le fait que si a et b sont deux entiers naturels non nuls, $\text{pgcd}(a, b) = \text{pgcd}(b, a \bmod b)$.

Pseudo Code

```

Data :  $a, b \in \mathbb{N}^*, a < b$ 
 $x \leftarrow a$ 
 $y \leftarrow b$ 
tant que  $y \neq 0$  faire
     $r \leftarrow$  reste de la division euclidienne de  $x$  par  $y$ 
     $x \leftarrow y$ 
     $y \leftarrow r$ 
fin
    
```

Exemple

Calculons le PGCD de 240 et 64 :

$$240 = 3 \cdot 64 + 48$$

$$64 = 1 \cdot 48 + 16$$

$$48 = 3 \cdot 16 + 0$$

$\text{pgcd}(240, 64)$ est le dernier reste non nul à savoir 16.

Démonstration

Soient a et b , tels que $(a; b) \in \mathbb{N}^*$ tels que $a > b$. Soient $(q; r) \in \mathbb{N}$ tels que $a = b \cdot q + r$. Avec r le reste de la division euclidienne de a par b et q le quotient.

Montrons que $\text{PGCD}(a, b) = \text{PGCD}(b, r)$.

Soit $d \in \mathbb{N}$ un diviseur commun de a et b . Dans ce cas, d divise $a - b \cdot q$; donc d divise r .

En conclusion, si d divise a et b alors $\text{PGCD}(a, b) = \text{PGCD}(b, r)$.

Réciproquement, soit $d \in \mathbb{N}$ un diviseur commun de b et r . Dans ce cas, d divise $b \cdot q + r$; donc d divise a .

En conclusion, si d divise b et r alors $\text{PGCD}(a, b) = \text{PGCD}(b, r)$.

On a donc $\text{PGCD}(a, b) = \text{PGCD}(b, r)$.

Démonstration

Preuve par invariant de boucle de l'algorithme d'Euclide

1. Initialement, a et b sont des données saisies par l'utilisateur. On a $x_0 = a$ et $y_0 = b$.
2. On postule que l'invariant de boucle est donné par $x = y \cdot q + r \iff r = x - y \cdot q$.
3. – Lors de la première itération, r_1 est tel que $r_1 = x_1 - y_1 \cdot q_1$.
– Lors de la nième itération, r_n est tel que $r_n = x_n - y_n \cdot q_n$.
– À l'itération suivante, $x_{n+1} = y_n$ et $y_{n+1} = r_n$, on peut donc trouver r_{n+1} tel que $r_{n+1} = x_{n+1} - y_{n+1} \cdot q_{n+1}$.
4. On a montré précédemment que l'itération se terminait.
5. En sortant de la boucle, l'invariant reste vrai.

Références

- [1] François Denis <http://pageperso.lif.univ-mrs.fr/~francois.denis/algoL2/chap1.pdf>
- [2] Alain Soyeur <http://asoyeur.free.fr/>
- [3] Wack et Al., *L'informatique pour tous en classes préparatoires aux grandes écoles*, Editions Eyrolles.
- [4]