

Chapitre 1 : Diversité et unité des langages de programmation

EXERCICES - Corrections

Maintenant que vous connaissez les spécificités de certains langages, nous allons pouvoir traduire un programme (ou algorithme) d'un langage à un autre. Nous allons nous concentrer sur des langages qui pourront être réutilisés ou tout du moins qui sont assez utilisés actuellement.

Pour écrire et tester les programmes, vous pouvez utiliser les éditeurs en ligne suivants (même s'ils ne sont pas indispensables pour la réalisation de l'activité) :

pour C : <https://repl.it/languages/c>
 pour Python : <https://repl.it/languages/python3> ou <https://www.codabrainy.com/python-compiler/>
 pour PHP (serveur web) : <https://repl.it/languages/php7> ou <http://www.writephponline.com/>
 pour Java : <https://repl.it/languages/java10>
 pour Javascript : directement dans la console d'un navigateur (Ctrl+Shift+K pour Firefox)

Exercice 1 (langage PHP)

Voici deux programmes en **PHP**, analysez les (et testez les éventuellement avec un éditeur en ligne).

Rappel : chaque variable en PHP commence par un \$ et chaque instruction se termine par un ;

1. Proposez le code Python correspondant à chaque programme.

```
<?php
// Ci-dessous, la fonction qui calcule le volume du cône
function VolumeCone($rayon, $hauteur)
{
    $volume = $rayon * $rayon * 3.14 * $hauteur * (1/3); // calcul du volume
    return $volume; // indique la valeur à renvoyer, ici le volume
}

$volume = VolumeCone(3, 1);
echo 'Le volume d'un cône de rayon 3 et de hauteur 1 est de ' . $volume;
?>
```

En Python :

```
def Volumecone(rayon, hauteur):
    volume = 3.14*hauteur/3*rayon**2
    return volume

volume = Volumecone(3,1)
print("Le volume d'un cône de rayon 3 et de hauteur 1 est de ", volume)
```

```
<?php
for ($nombre_de_lignes = 1; $nombre_de_lignes <= 100; $nombre_de_lignes++)
{
    echo 'Ceci est la ligne n°' . $nombre_de_lignes . '<br />';
}
?>
```

En Python :

```
for nombre_de_lignes in range(1,101):
    print("Ceci est la ligne n°", nombre_de_lignes)
```

2. Maintenant proposez le code d'une fonction Python `volume_sphere(r)` permettant de calculer et afficher le volume d'une sphère de rayon r . (pour rappel le volume d'une sphère est $V_{\text{sphère}} = \frac{4}{3} \times 3,14 \times r^3$)

```
def Volume_sphere(r):
    volume = 4/3*3.14*r**3
    return volume
```

3. Traduisez cette fonction en une fonction PHP.

```
<?php
// Ci-dessous, la fonction qui calcule le volume d'une sphère de rayon r
function Volume_sphere($rayon)
{
```

```

$volume = $rayon * $rayon * $rayon * 3.14 * (4/3); // calcul du volume
return $volume; // indique la valeur à renvoyer, ici le volume
}
?>

```

4. Proposez le code **Python** permettant d'afficher le volume des sphères de rayons 1 à 10 par pas de 1.

```

def Volume_sphere(r):
    volume = 4/3*3.14*r**3
    return volume
for r in range (1,11):
    print("Le volume de la sphère de rayon ", r, " est ", Volume_sphere(r))

```

5. Écrivez le code PHP correspondant.

```

<?php
// Ci-dessous, la fonction qui calcule le volume d'une sphère de rayon r
function Volume_sphere($rayon)
{
    $volume = $rayon * $rayon * $rayon * 3.14 * (4/3); // calcul du volume
    return $volume; // indique la valeur à renvoyer, ici le volume
}

for ($rayon = 1; $rayon <= 10; $rayon++)
{
    $volume = Volume_sphere($rayon);
    echo 'Le volume de la sphère de rayon ' . $rayon . ' est ' . $volume . '<br />';
}
?>

```

Exercice 2 (langage Javascript)

Voici un programme écrit dans le langage **Javascript**.

```

function tri(tab){
    for (var i = 0; i < tab.length; i++){
        var ind_min = i;
        for (var j = i+1; j < tab.length; j++){
            if (tab[j] < tab[ind_min]){
                ind_min = j;
            }
        }
        var temp = tab[i];
        tab[i] = tab[ind_min];
        tab[ind_min] = temp;
    }
    return tab;
}

var t = [2, 6, 1, 6, 3, 10];
tri(t);
console.log(t);

```

En Python :

```

def tri_selection(tab):
    for i in range(len(tab)):
        # Trouver le min
        ind_min = i
        for j in range(i+1, len(tab)):
            if tab[j] < tab[ind_min]:
                ind_min = j

        tmp = tab[i]
        tab[i] = tab[ind_min]
        tab[ind_min] = tmp
    return tab

t = [2, 6, 1, 6, 2, 3, 10]

tri_selection(t)
print ("Le tableau trié est:" , t)

```

1. Quel algorithme reconnaissez-vous ? **L'algorithme de tri par sélection.**

Vous pouvez tester ce code dans un navigateur : avec Firefox, lancez la console avec Ctrl+Shift+K et écrivez le code dans la fenêtre qui s'ouvre puis cliquez sur « Exécuter ».

2. Traduisez cet algorithme en Python. (**cf tableau**)

3. Ecrivez en Javascript l'algorithme de tri par insertion.

```
function tri_insertion(tab){
  for (var i = 1; i < tab.length; i++){
    var x = tab[i] ;
    var j = i ;
    while ( j > 0 && x < tab[j-1] ){
      tab[j] = tab[j-1] ;
      j = j-1;
    }
    tab[j] = x ;
  }
  return tab;
}

var t = [2, 6, 1, 6, 3, 10];
tri_insertion(t);
console.log(t);
```

En Python tri par insertion (cf séquence 5):

```
def tri_par_insertion(tab):
  for i in range(1,len(tab)):
    x = tab[i]
    j = i
    while j>0 and x < tab[j-1]:
      tab[j] = tab[j-1]
      j = j -1
    tab[j] = x
  return tab

tab = [2, 6, 1, 6, 2, 3, 10]

tri_insertion(tab)
print ("Le tableau trié est:" , tab)
```

Exercice 3 (langage C)

Voici un programme en C , analysez le, (testez le éventuellement avec un éditeur en ligne) et traduisez le en python après avoir lu les explications qui suivent.

```
#include <stdio.h>
int main(void) {
  int i;
  i=19;
  if (i<18){
    printf("Vous êtes mineur");
  }
  else {
    printf("Vous êtes majeur");
  }
  return 0;
}
```

En Python :

```
i = 19

if i < 18 :
    print("vous êtes mineur")
else :
    print("vous êtes majeur")
```

Détaillons ce script en langage C :

- La première ligne du programme : `#include <stdio.h>` est une "directive de préprocesseur", cette ligne est lue au début de la phase de compilation. Vous devez juste savoir que cette ligne est obligatoire si vous voulez utiliser "printf" (voir quelques lignes plus loin).
- Le programme commence réellement avec la ligne : `int main(void) {`
- Nous avons ici une fonction (équivalent à un "def main():" en Python), comme vous l'avez sans doute déjà compris, tout le code compris entre l'accolade ouvrante "{" et l'accolade fermante "}" constitue la fonction nommée "main". le code constituant une fonction n'est pas défini grâce à une indentation comme en python, mais grâce à des accolades (vous remarquerez que l'on utilise tout de même une indentation, mais cette dernière n'est pas obligatoire en C mais seulement fortement conseillée afin de rendre le code plus lisible).
- Le "int" situé avant le nom de la fonction ("main" dans cet exemple), signifie que la fonction doit renvoyer un entier. Le "void" situé entre parenthèses signifie que notre fonction ne prend aucun paramètre.
- La ligne : `printf("...");` permet d'afficher le message "..." dans la console
- Le `return 0;` permet de renvoyer l'entier 0. Pourquoi renvoie-t-on 0 ? Tout simplement pour signifier que la fonction "main" a bien été exécutée jusqu'au bout. Si cette fonction renvoie autre chose que 0 cela signifiera qu'il y a eu un problème lors de l'exécution de cette dernière.
- Vous avez dû remarquer que la plupart des lignes se terminent par un point-virgule ;. Ce point virgule indique au compilateur que la ligne est terminée, il est obligatoire.

Exercice 4 Optionnel (langage Java)

Voici un programme écrit en **Java**

```
import java.util.*;
import java.lang.Math;

public class JavaApplication {
    private static Scanner input = new Scanner(System.in);

    public static void main(String[] args) {
        int a;
        int b;
        System.out.println("Enter two numbers. This algorithm will find the greatest common denominator.");
        a = input.nextInt();
        b = input.nextInt();
        while (!(b == 0)) {
            if (a > b) {
                a = a - b;
            } else {
                b = b - a;
            }
        }
        System.out.println(a);
    }
}
```

1. Quelles sont les variables d'entrée et de sortie de ce programme ?

Les variables d'entrées sont des entiers a et b ; la sortie de ce programme est un entier a

2. Quel type de boucle est utilisée ?

La boucle utilisée est une boucle tant que (while)

3. A quoi sert ce programme ?

Ce programme permet de trouver le plus grand dénominateur commun entre deux entier

4. Proposez une traduction en Python de ce programme.

```
print("Enter two numbers. This algorithm will find the greatest common denominator.")
a = int(input())
b = int(input())
while not b == 0:
    if a > b:
        a = a - b
    else:
        b = b - a
print(a)
```

5. Proposez une traduction en PHP puis en Javascript (attention les valeurs de a et b doivent être définies dans le programme et non demandées à l'utilisateur.)

en PHP :

```
<?php
# The following function is called whenever your program needs to input data.
# You can change it to use other sources rather than fgets().
function input() {
    return fgets(STDIN);
}
```

```

echo "Enter two numbers. This algorithm will find the greatest common denominator." . PHP_EOL;
$a = input();
$b = input();
while (!( $b == 0)) {
    if ( $a > $b) {
        $a = $a - $b;
    } else {
        $b = $b - $a;
    }
}
echo $a . PHP_EOL;
?>

```

en Javascript

```

function main() {
    var a;
    var b;

    window.alert("Enter two numbers. This algorithm will find the greatest common denominator.");
    a = window.prompt('Enter a value for a');
    b = window.prompt('Enter a value for b');
    while (!(b == 0)) {
        if (a > b) {
            a = a - b;
        } else {
            b = b - a;
        }
    }
    window.alert(a);
}

```

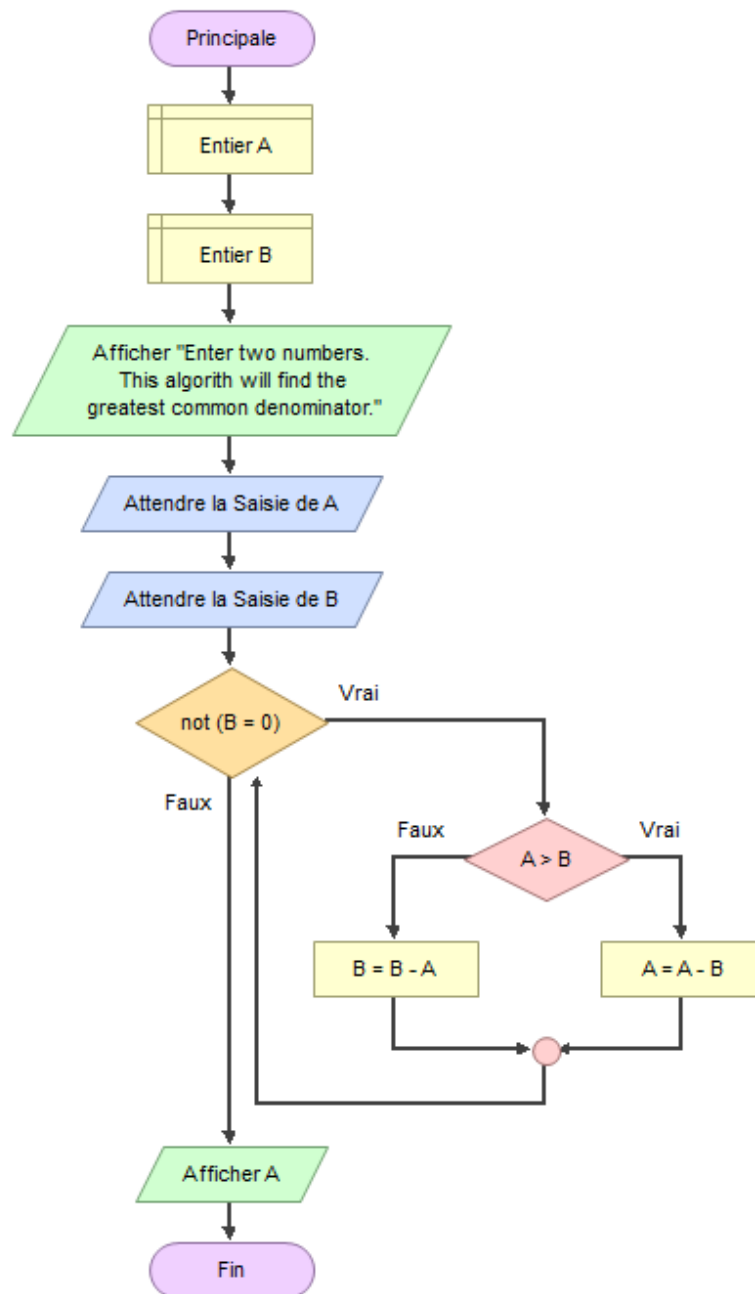
Remarques complémentaire : le logiciel Flowgorithm

téléchargeable à l'adresse: <http://www.flowgorithm.org/>

Ce logiciel permet de créer des 'algorigrammes' (voir page suivante l'algorigramme du programme précédent) Un algorigramme permet de schématiser en utilisant un organigramme l'algorithme d'un programme, ce qui permet de visualiser schématiquement le déroulé de l'algorithme et de mieux comprendre son fonctionnement. L'autre intérêt de Flowgorithm est qu'il permet de 'traduire' dans différents langages (Python, C++, C#, PHP, Javascript...) ainsi que le pseudocode de l'algorithme (voir page suivante)

Sur le site de Flowgorithm vous trouverez en plus de la documentation du logiciel, plusieurs exemples déjà compilés pour les comparer.

(nb : Flowgorithm est adapté pour les fonctions simple ou les programmes courts, il ne peut pas gérer les algorithmes trop complexes...)


Pseudo code de l'algorithme d'Euclide du plus grand dénominateur commun :

Fonction Principale

Déclarer la Variable A de Type Entier

Déclarer la Variable B de Type Entier

Afficher "Enter two numbers. This algorithm will find the greatest common denominator."

Attendre la Saisie de A

Attendre la Saisie de B

Tant Que not (B = 0)

Si A > B

Assigner à la Variable A l'Expression A - B

Faux:

Assigner à la Variable B l'Expression B - A

Fin

Fin

Afficher A

Fin