

Programmation java

La boucle for

Pierre Ortegat

EPHEC

1. La boucle for

2. Exercices

La boucle for

La boucle for peut se “traduire” comme suit en français:

Après avoir initialisé la boucle; tant que la condition est vraie;
exécuter les instructions de la boucle. Après chaque itération, avant
de vérifier la condition; appliquer l’incrément.

for: syntaxe

Après **<initialisation>** de la boucle; tant que la **<condition>** est vraie; exécuter les **<instructions de la boucle>**. Après chaque itération, avant de vérifier la condition, appliquer l' **<incrément>**.

Pseudo-code:

```
1  for( <initialisation> ; <condition> ; <incrément> ){  
2      <instructions de la boucle>  
3  }
```

for: exemple

Pseudo-code:

```
1  for( <initialisation de la variable> ; <condition> ; <  
    incrément> ){  
2    <instructions de la boucle>  
3  }
```

Exemple:

```
1  for( int i =0 ; i < 10 ; i++ ){  
2    System.out.println(i);  
3  }
```

while vs for

```
1  int i = 0;  
2  
3  while(i <= 10) {  
4      System.out.println(i);  
5      i++;  
6  }
```

```
1  for( int i =0 ; i <= 10 ; i++ ){  
2      System.out.println(i);  
3  }
```

Exercices

Écrire un programme qui demande à l'utilisateur 10 nombres et, après que l'utilisateur les aie entrés; affiche le plus grand.

Avant chaque entrée, dire à l'utilisateur il en est à combien de nombres entrés sur 10.

Écrire un programme qui demande cinq entiers positifs à l'utilisateur. Le programme affichera ensuite la moyenne et la somme de ces nombres.

Contrainte: écrire deux variantes du programme, une avec "while" et l'autre avec "for"

Écrire un programme qui demande deux entiers à l'utilisateur et affiche en retour le premier nombre mis à la puissance du second.

Par exemple, si l'utilisateur entre 2 et 5, le programme devra afficher 32 car

$$2^5 = 32$$

Contraintes: ne pas utiliser de mécanismes non-vu au cour (aka: la lib math) et supporter des valeurs de résultats jusqu'à, au minimum, 2^{60}

Écrire un programme qui demande des nombres entiers à l'utilisateur. Dès que l'utilisateur entre un nombre plus grand ou égal au cube du premier nombre, afficher la différence entre la somme de tous les nombres entrés et le dernier.

Note: le cube d'un nombre peut se définir comme ceci:

$$\text{cube}(x) = x^3 = x * x * x$$

Contrainte: mettre la résolution de l'exercice précédent dans une fonction et utiliser celle ci pour calculer les cubes.

Écrire un programme qui demande un nombre entier à l'utilisateur.
Ce programme affichera en retour la somme de nombres entiers
entre zéro et le nombre entier.

*Attention ! Le nombre entré peut être négatif **et** valide !*

while vs for VI

Écrire un programme de caisse enregistreuse qui demande à l'utilisateur d'encoder tous les prix des articles achetés, les un à la suite des autres.

Lorsque l'utilisateur entre 0; cela signifie qu'il a fini d'encoder ses articles. Le programme affichera donc le montant total.

Le programme demandera ensuite à l'utilisateur d'encoder la somme en liquide donnée par le client.

Enfin, le programme affichera deux choses:

- la somme à rendre
- comme composer cette somme avec le moins de pièces de 1, 2 et de billets de 10 (dans cet exercices, seul ces valeurs monétaires existent).

Faire bien attentions à tous les cas particuliers !