

Boucles, fonctions, surcharge, varargs

Exercice 1 - Table de multiplication

Afficher la table de multiplication de 1 à 10, c-a-d une matrice de 10x10 avec les valeur de 1 à 10 en ligne et en colonne et dans chaque case le produit de la ligne par la colonne.
Modifier le programme pour que l'utilisateur choisisse la taille de la table.

Exercice 2 - Un triangle d'étoile

On souhaite afficher un triangle d'étoiles dont la taille est demandée au préalable à l'utilisateur.

Pour 4 cela donne :

```
*
**
***
****
```

Faire maintenant la même chose mais en dessinant une pyramide.

Exercice 3 - Maximum/minimum de deux nombre

Dans la classe `Numerical`

Ecrire une fonction `max` qui prend deux entiers en paramètre et renvoie le maximum des deux entiers.

Ecrire un `main` de test.

Ecrire une fonction `min` qui prend deux entiers en paramètre et renvoie le minimum des deux entiers.

Modifier la méthode `main` pour tester. aussi le minimum.

Ecrire une fonction `max` qui prend deux nombres à virgule en paramètre et renvoie le maximum des deux nombres.

Modifier le `main` pour tester.

Exercice 4 - Somme des entiers consécutifs

On souhaite calculer la somme des entiers de 1 à `n`.

Dans un premier temps tout faire dans le `main`

Dans un second temps, écrire une méthode statique `sum` qui calcul la somme des entiers et appelée celle-ci de la méthode `main`

Exercice 5 - Factoriel

On souhaite calculer la valeur de factoriel `n` pour un `n` donnée par l'utilisateur.

Écrire une méthode `fact` permettant de réaliser ce calcul. Notons que $fact(n) = fact(n-1) * n$ c-a-d l'ancienne valeur de factoriel à laquelle on multiplie `n` et que $fact(0) = 1$.

À partir de quel valeur, le cacul effectue t'il un dépassement de capacité ; vous pouvez tester

avec des `int` puis avec des `long`.

Exercice 6 - Afficher les arguments de la ligne de commande

Écrire une classe `PrintArgs` qui affiche les arguments de la ligne de commande.

```
$ java Voici des arguments
Voici
des
arguments
```

Les arguments de la lignes de commandes sont stockées dans le tableau de chaînes de caractères passé en argument à la méthode `public static main(String[] args)`.

Dans un premier temps, écrivez une boucle de parcours en sachant qu'en Java les tableaux possède un attribut `length` qui renvoie la taille du tableau.

Écrire la même boucle en utilisant la notation "foreach" i.e. `for(type variable:tableau)`

Exercice 7 - Chaîne de caractère

On souhaite effectuer différentes manipulations sur les chaînes de caractère.

Écrire une méthode qui prend en paramètre une chaîne de caractère, découpe celle-ci en mot et met un mot sur deux en majuscule minuscule respectivement.

On utilise pour cela `split()`, `toUpperCase` et `toLowerCase()`.

Écrire une méthode `mirror` qui renvoie une chaîne de caractère quise en paramètre à l'envers ("toto" devient "otot") en utilisant `charAt()` et la concaténation de chaîne de caractère (+).

Ré-écrire la méthode `mirror` en utilisant un `StringBuilder` et sa méthode `append` pour effectuer la même opération.

Quelle implantation est la plus efficace ?

Exercice 8 - En morse. Stop.

Écrire une classe `Morse` qui permet lors de son exécution d'afficher les chaînes de caractères prisent en argument séparé par des "Stop. ".

```
$ java Morse ceci est drole
ceci Stop. est Stop. drole Stop.
```

Exercice 9 - Somme d'entiers

- 1 Écrire une méthode `sum` qui calcule la somme d'un certain nombre d'entier passés en paramètre de la méthode.
- 2 Écrire une méthode `main` de test.
- 3 Pourquoi peut-on déclarer le tableau pris en paramètre comme un `int...` et non comme un `int` ?
Modifier la méthode `sum` et la méthode `main` en conséquence.

Exercice 10 - Conversion de String en entier

Écrire un programme qui met les nombres entiers qu'on lui passe en argument sur sa ligne de commandes dans un tableau d'entiers et qui affiche ce tableau suivi de la somme de ses éléments. Voici un exemple d'exécution :

```
$ java Sum 15 5 231
15 5 231
Somme: 251
```

La méthode statique `parseInt(String s)` de la classe `Integer` permet de récupérer la valeur d'un entier stockée dans un chaîne de caractères.

Que se passe-t'il lorsqu'une lettre est prise en argument au lieu d'un nombre ?

Exercice 11 - Tri à bulle

Écrire une méthode `swap` qui échange les valeurs de deux cases d'un tableau. `void swap(int[] array, int index1, int index2)`

Écrire une méthode `indexOfMin` qui renvoie l'index de la valeur minimal d'un tableau.

Modifier la méthode `indexOfMin` en ajoutant deux index indiquant que l'on cherche l'index du minimum non pas sur tout le tableau mais sur la partie de tableau entre les deux index (le premier compris, le deuxième non compris).

Écrire la méthode `sort` qui prend un tableau d'entier en paramètre et qui trie celui-ci en utilisant pour cela les méthodes `indexOfMin` et `swap`.