

# Introduction à la programmation et à la modélisation objet en Java

Jean-François Viaud

January 6, 2020

## 1 Sous-programmes

- Notions
- Procédures
- Fonctions

Sous-programme : Une partie de programme résolvant une partie de problème. Un sous-programme équivaut à une nouvelle instruction du langage.

Utilités :

- Réutilisation
- Lisibilité
- Décomposition d'un problème complexe

- Procédure, fonction, méthode, interface
- Appel, Paramètres : formels et effectifs, Passage par valeur ou référence, Retour
- Effets de bord, Visibilité

Une procédure n'a pas de valeur de retour, elle n'a que des effets de bord.

Exercices :

- 1 Ecrire puis utiliser un sous-programme qui affiche les nombres compris entre 1 et 100
- 2 Compléter en affichant la somme de ces nombres
- 3 Montrer le schéma de communication et le déroulement de l'exécution. (Utilisation du débogueur)
- 4 Modifier la procédure précédente en passant les bornes en paramètre.
- 5 Utiliser de ce nouveau sous-programme et montrer à nouveau le schéma de communication et le déroulement de l'exécution.

- En Java, une fonction ne retourne qu'un seul objet.
- Et dans tous les langages, une fonction ne devrait pas avoir d'effet de bord.
- Cohérence entre le type de résultat à retourner et le type de la valeur effectivement retournée.
- En particulier, obligation de retour pour les fonctions.
- Aucune instruction se trouvant après le return n'est exécutée.

- 1 Modifier le sous-programme précédent pour que l'affichage se fasse dans le main.
- 2 Ecrivez une fonction `long Factorielle(int n)`
- 3 Test UNITAIRES et COMMENTAIRES (+ javadoc) : 1 ligne de code = 1 ligne de commentaire + 1 ligne de test. Completez le source en conséquence.

Remarques :

- `Factorielle(4)` est une expression dont la valeur vaut 24.
- Cette valeur peut être affectée à une variable `a = Factorielle(4);`, testée `if(Factorielle(4)>x){...}`, ou fournie à un autre sous-programme `Dialogue.afficher(Factorielle(4));`.

