

# Projet de programmation orientée objet en c++

## Partie 2

INSA CVL, MRI (4A)

18 octobre 2021

Nous allons surcharger quelques opérateurs pour nos classes `Tache` et `ListeDeTaches`. Il est évidemment possible (et conseillé) de réutiliser les fonctions de classe que vous avez implémentées dans la partie 1 dans le code de vos surcharges d'opérateurs.

### Exercices

**Exercice 1.** Implémenter un constructeur par copie à la classe `ListeDeTaches`.

**Exercice 2.** Surcharger l'opérateur de sortie `<<` pour chacune des deux classes. L'affichage d'une tâche devra être similaire à celui de l'exemple suivant :

Mise à jour : Mettre à jour le logiciel (priorité : 5)

L'affichage d'une liste de tâches devra être similaire à celui de l'exemple suivant :

Liste de 4 tâche(s) :

- 1) Mise à jour : Mettre à jour le logiciel (priorité : 5)
- 2) Achat : Acheter des pièces de rechange (priorité : 4)
- 3) Panne : Réparer la machine 856 (priorité : 8)
- 4) Maintenance : Vérifier la machine 364 (priorité : 1)

**Exercice 3. dans la classe `Tache`** : Il serait pratique de pouvoir utiliser directement les opérateurs de comparaison (`==`, `!=`, `<=`, `>=`, `<` et `>`) sur les tâches pour comparer leur priorité. Surcharger ces opérateurs.

**Exercice 4. dans la classe `ListeDeTaches`** : Il serait pratique de pouvoir accéder directement aux tâches d'une liste via l'opérateur `[]`. Surcharger cet opérateur pour que l'instruction `l[i]` renvoie la  $i^{\text{ème}}$  tâche de la liste `l`.

**Exercice 5. dans la classe `ListeDeTaches`** : Surcharger l'opérateur d'accumulation pour qu'il permette d'ajouter une tâche à une liste : par exemple, en considérant une tâche `t` et une liste `l`, l'instruction `l+=t` ajoutera la tâche `t` à la liste `l`.

**Exercice 6. dans la classe `ListeDeTaches`** : Surcharger l'opérateur d'incrémentation pour qu'il permette de traiter et de retirer la tâche de plus haute priorité dans la liste : par exemple, en considérant une liste `l`, l'instruction `l--` traitera et supprimera la tâche de plus haute priorité de la liste, et renverra un booléen indiquant si le traitement s'est bien passé ou non.

**Exercice 7.** Faire en sorte d'utiliser les opérateurs (lorsque c'est pertinent) dans le code de vos fonctions de classe et de la fonction `main`. Si nécessaire, ajouter quelques tests dans le `main` pour vérifier que tout fonctionne bien.