

STI 4^e année – Java avancée

TP

18 novembre 2020

Le problème des lecteurs-rédacteurs

Ceci est un problème classique d'accès concurrents à une ressource partagée par deux types d'entités :

- des lecteurs qui accèdent à la ressource sans la modifier. Plusieurs lecteurs peuvent y accéder en même temps.
- des rédacteurs qui, eux, peuvent modifier la ressource. L'accès pour les rédacteurs est un accès exclusif. En d'autres termes, si un rédacteur travaille avec la ressource, aucune autre entité (lecteur ou rédacteur) ne doit accéder à celle-ci.

Ce schéma est typiquement utilisé pour la manipulation de fichiers ou de zones mémoires.

On considère le cas simple où :

- la ressource partagée est une chaîne de caractère : classe Ressource doit la gérer ;
- le rédacteur met à jour la ressource en y mettant son nom : classe Redacteur. Sa méthode `run()` doit exécuter $n=10$ itérations de l'algorithme suivant :

```
1 Prendre la ressource ;
2 Modifier la ressource ;
3 Libérer la ressource ;
4 Attendre 3s ;
```

- le lecteur lit la ressource et l'affiche en indiquant son nom : classe Lecteur. Sa méthode `run()` doit exécuter $n=5$ itérations de l'algorithme suivant :

```
1 Lire la ressource ;
2 Afficher sa valeur ;
3 Attendre 5s ;
```

Exercice 1 Implémenter une solution java pour ce problème dans le cas où :

- La priorité aux rédacteurs : un rédacteur demandant à écrire sera autorisé à le faire dès que tout les lecteurs et/ou rédacteurs actuels auront fini leur accès (i.e. aucun nouveau lecteur n'est admis si un rédacteur désire écrire). Pour tester, on crée 5 lecteurs et trois rédacteurs.
- Le nombre de lecteurs borné : on tolère qu'un nombre m de lecteurs qui peuvent accéder à la ressource simultanément. Pour tester, on crée $2m$ lecteurs et m rédacteurs.

Exercice 2 Implémenter une solution java où on ne peut instancier qu'un seul rédacteur à la fois.