UFR IM²AG

UNIVERSITÉ Grenoble Alpes

Master M1 Informatique, CSE
Année 2020-2021

Résumé

Ce TD/TP vise à améliorer votre compréhension des problèmes de synchronisation et de leurs solutions. Vous y aurez l'occasion d'utiliser la partie des primitives fournies par la bibliothèque de *threads* POSIX dédiées à la synchronisation de threads (`pthread_mutex_...()`, `pthread_cond_...()`).

Vous aurez probablement besoin de consulter les pages de manuel associées (par exemple `man 3 pthread_mutex_lock`) afin de prendre connaissance de leur sémantique précise (Remarque : il y a beaucoup d'autres fonctions, et leur page de manuel associée, dans la bibliothèque `pthread`. Pour en obtenir une liste complète, exécutez : `man -k pthread`).

Le barbier

La boutique du barbier est composée d'une salle d'attente, contenant N chaises, et du salon où se trouve la chaise du barbier. Lorsque le barbier a fini de raser un client, il fait entrer le client suivant dans le salon. Si la salle d'attente est vide, le barbier s'y installe pour dormir. Si un client trouve le barbier endormi, il le réveille. Sinon, il s'installe dans la salle d'attente s'il reste de la place (et rentre chez lui sinon). Les clients passent devant le barbier dans l'ordre de leur arrivée bien sûr.

Le but est de vous faire modéliser le problème et d'implanter les synchronisations associées. Vous êtes libres dans le choix des structures de synchronisation et des variables partagées (d'état).

Pour arriver à la solution finale, vous pouvez considérer plusiers variantes :

I Ne pas gérer l'ordre d'arrivée des clients et faire attendre les clients même si la salle d'attente est pleine (ils seront dehors au froid !)

II Ne pas gérer l'ordre d'arrivée des clients mais intégrer la contrainte des limite de places dans la salle d'attente.

III Gérer l'ordre FIFO.

Pour chaque variante de problème vous auriez à :

Questions :

1. Modéliser les synchronisations de votre programme, en particulier, autour des ressources communes à partager et des évènements représentant des changements d'états de votre modélisation.
2. Implanter votre modélisation. Afficher une trace (printf) pour chacun des évènements.

3. Vérifier que la trace est correcte : tous les évènements ont bien lieu dans l'ordre attendus.