

ASR TD6

Threads, synchronisation

Atomique !

Exercice 1 Section critique Soient 2 threads T1 et T2 effectuant les calculs suivant sur des variables globales :

```
unsigned int x = 1;
unsigned int y = 1;

void* T1() {
    x+=1;
    y+=1;
}

void* T2() {
    x*=2;
    y*=2;
}
```

1. L'exécution de ces threads peut produire différents résultats. Donner différents résultats possibles.
2. Mettre en place une section critique à l'aide d'un mutex pour garantir que les opérations de chaque thread seront effectuées de manière atomique. La mise en place de cette section critique doit garantir qu'à la fin de l'exécution $x = y$.

Exercice 2

Liste chaînée

1. Créer une liste chaînée de 20 éléments.
2. Créer un thread supprimant les 5 premiers éléments de la liste.
3. Créer un autre thread ajoutant 5 éléments à la liste.
4. Lancer les 2 threads. Que se passe-t'il ou que risque t'il de se passer ?
5. Proposer une solution basée sur la mise en place d'une section critique.

Exercice 3

Aiguillage

Soit l'aiguillage suivant :

1. Chaque train est modélisé par un thread différent. Que se passe-t-il lorsque plusieurs trains souhaitent accéder à la portion partagée ?
2. Proposer une solution basée sur l'utilisation d'un mutex.

Accès concurrents

Exercice 4 Salle d'attente

Soit une salle d'attente comportant 8 places et des personnes cherchant à entrer dans cette salle. Seules 8 personnes sont autorisées à rentrer dans la salle. Si la salle est pleine, les personnes restent en attente à l'extérieur jusqu'à ce que des places se libèrent. Les personnes dans la salle d'attente en sortent au bout d'un temps variable.

1. Mettre en place des threads représentant le comportement des personnes ainsi qu'un sémaphore permettant de limiter le nombre de personnes dans la salle d'attente à 8.
2. Un problème avec cette implantation est que plusieurs personnes peuvent entrer ou sortir simultanément dans la salle d'attente, or la porte ne permet de ne laisser passer qu'une personne à la fois ! Ajouter un mutex (à placer correctement) afin de résoudre ce problème.