

## TD9 : Threads (2)

### Exercice 1 : Mécanisme d'exclusion mutuelle

Ecrire un programme qui crée  $N$  threads ayant chacun pour but d'incrémenter le même compteur et ceci dix (10) fois chacun. Ce compteur doit donc être protégé par une section critique.

### Exercice 2 : Mécanisme de Condition

L'objectif de l'exercice est de faire passer un message (entier) d'un thread à l'autre en passant par  $N$  interlocuteurs. Chaque interlocuteur  $i$  peut transmettre/recevoir un message à/de ses voisins directs ( $i-1$ ) et ( $i+1$ ).

Soit un nombre choisi aléatoirement entre 1 et 100 par le thread 1, faites-en sorte que le dernier thread le reçoive et l'affiche. Vous utiliserez le mécanisme des *Conditions* pour gérer l'attente des threads.

### Exercice 3 : Mécanisme de barrière

L'objectif de l'exercice est d'offrir à l'utilisateur un mécanisme de barrière entre  $N$  threads réutilisable, ceci en utilisant uniquement des Locks et des *Conditions*. Il est proposé de l'implémenter en suivant plusieurs étapes :

- 1) La synchronisation est effectuée sur 3 threads, les deux premiers attendent que le troisième les libère.
- 2) Les  $(n-1)$  premiers threads attendent que le dernier les libère.