

# Théorie et pratique de la concurrence – Master 1 Informatique

## TP 4 : File FIFO partagée en C

### Exercices

**Exercice 1:** *Producteurs-consommateurs avec FIFO et variables de condition*

Le but est de reprendre l'exercice 3 du TP 2 sur les producteurs consommateurs. Pour rappel on dispose d'une file FIFO partagée entre des producteurs et des consommateurs. Le nombre de consommateur est connu. On souhaite que tous les consommateurs aient lu un élément de la file avant de le retirer. On veut de plus interdire que deux consommateurs lisent en même temps la même donnée sur la file (c'est-à-dire les lectures sont protégées par des verrous). On proposera trois implémentations différentes.

1. Dans la première implémentation, un consommateur pourra lire des données quand elles sont disponibles (*Indication : il faut ajouter des variables de condition pour la communication entre les producteurs et les consommateurs*). Pour cette implémentation, deux consommateurs ne pourront pas lire en même temps à deux endroits de la file (c'est-à-dire la file est protégée par un verrou global). Toutefois, chaque consommateur libérera régulièrement le verrou sur la file (typiquement après avoir lu une donnée).
2. La deuxième n'autorisera pas un consommateur à lire une nouvelle donnée tant que la dernière donnée lue n'aura pas été vue par tous les consommateurs (le dernier consommateur aura donc pour rôle de réveiller les autres consommateurs et de supprimer les éléments de la file).
3. La troisième implémentation étendra la première, on voudra de plus que plusieurs consommateurs puissent lire en même temps différents endroits de la file (mais deux consommateurs ne pourront pas lire en même temps le même endroit de la file). *Indication : il faut associer un verrou différent à chaque élément de la file.*

### Remarques :

- Dans les trois cas, si une nouvelle donnée n'est pas disponible pour un consommateur il devra s'endormir et sera réveillé par le producteur.
- Il faut faire attention aux cas limites et en particulier aux pointeurs que vous manipulez. En particulier faire attention à la manipulation de pointeurs en concurrence. Par exemple, si un consommateur fait un free sur une cellule pointée par un autre consommateur.