

# Programmation Système

## TP 8 : tubes, FIFO

Juliusz Chroboczek  
Khouloud Zine Elabidine

19 novembre 2012

**Exercice 1** (Grep parallèle). Écrivez un programme `occurrences` qui prend en paramètre une expression régulière  $r$  et  $n$  noms de fichiers puis qui affiche le nombre total d'occurrences de  $r$  dans les  $n$  fichiers en exécutant  $n$  occurrences de « `grep -n` » en parallèle. Le père lira tous les résultats et calculera le nombre total d'occurrences.

**Exercice 2** (Gestion de logs). Le but de cet exercice est d'écrire un démon semblable au démon `syslog` tournant sur tout système Unix.

1. Écrivez un programme `log-daemon` qui commence par créer une FIFO nommée `/tmp/logger` accessible en écriture mais pas en lecture par tout utilisateur. Votre programme lira ensuite les données qui arrivent sur la FIFO, et les affichera sur sa sortie standard. Que se passe-t-il si vous lancez votre programme sans qu'aucun écrivain ait ouvert la FIFO ? Vérifiez votre hypothèse à l'aide de la commande `strace`.
2. Écrivez un programme `logger` qui formate ses paramètres de ligne de commande sur la FIFO `/tmp/logger` en les terminant par un caractère de fin de ligne. Testez votre programme. Que se passe-t-il si vous exécutez `logger` alors que `log-daemon` ne s'exécute pas ? Testez votre hypothèse à l'aide de la commande `strace`.
3. Modifiez le programme `log-daemon` pour qu'il reouvre la FIFO lorsque l'appel système `read` retourne 0.
4. Modifiez le programme `log-daemon` pour qu'il précède chaque ligne qu'il affiche de la date courante (vous pouvez vous servir des fonctions `localtime` et `strftime` avec le format `"%F %T"`).