

# UNIX - TD n° 5 : Execution de programmes externes

## 1 Premiers pas

1. Ecrire un programme en langage C exécutant la commande "ls -l" dans un processus fils (voir les fonctions `execxx()`). Le processus père attendra la fin de l'exécution du fils. Quelle est la commande `execxx()` que vous avez utilisée ? Pour quelles raisons ?
2. Écrire le même programme mais en utilisant un tube de communication pour que le processus père reçoive les résultats générés par le processus fils. Ces résultats seront affichés par le père et non par le fils.
3. À partir des questions précédentes, écrire un programme reproduisant l'exécution de la commande "ls -l | tail -3 | wc -w" en appelant chacune des trois commandes par un `execxx()` et dont les résultats sont affichés par le processus père. Dessinez l'état de la table des descripteurs de fichiers pour chacun des processus, au cours de l'exécution de ce programme.

## 2 Chien de garde

Reprenez un programme qui crée un processus fils réalisant un travail de longue durée. Réalisez un système de sécurité au niveau du processus père qui détruit le processus fils si son exécution dépasse les 5 secondes (voir les fonctions `alarm()` et `kill()`).

## 3 `execxx()` et les signaux

1. Un signal configuré par l'utilisateur (via la fonction `signal()`) peut-il être utilisé après l'exécution d'un programme par `execxx()` ? Écrivez un programme mettant en évidence ce comportement.
2. Un programme exécuté par `execxx()` peut-il être configuré pour qu'il ignore un signal (si oui, lesquels ?) ? Écrivez un programme mettant en évidence ce comportement.