

Programmation système

TD n° 1

Multi-processing

1 Exécution concurrente des processus père et fils

Écrire un programme qui crée un fils :

- Le père affiche : “Je suis le père mon PID est ... et mon PPID est ...”
- Le fils affiche : “Je suis le fils mon PID est ... et mon PPID est ...”

Rajouter l’appel à la primitive `sleep` pour voir les exécutions possibles :

1. le fils se termine avant le père,
2. le père se termine avant le fils.

2 Quelques attributs hérités et copie

Reprendre le programme ci-dessus et compléter en affichant l’uid, le gid, et le contenu d’une variable `x` initialisée à 2 (avant le fork) et modifié selon `x+3` par le fils et selon `x*5` par le père.

3 Cas particulier des descripteurs de fichiers

Écrire un programme qui ouvre un fichier nommé `toto`, en lecture et écriture, dont le contenu est la suite 123456789. Le programme fork ensuite ; le fils écrit `ab` dans le fichier ensuite il s’endort et il lit 2 caractères ; le père s’endort, lit 2 caractères et écrit `AB`.

4 Synchronisation des processus père et fils par la commande wait

Écrire un programme qui prenant une matrice carrée de taille 2 en paramètre et crée quatre fils. Chaque fils calcule un element du carré de la matrice initiale et le renvoi au processus père comme code de retour. Le père ensuite affiche la matrice.