

## TD5 : Processus

### Exercice 1

En utilisant la fonction `fork( )`, écrire les programmes décrivant les comportements suivants pour un processus père et son processus fils :

1. Affichage de 'Hello world !' par les deux processus
2. Affichage de 'Mon PID est ... et celui de mon père/fils est ... !'
3. Le processus fils choisit un nombre aléatoire entre 1 et 50, l'affiche, puis le communique à son père qui l'affiche à son tour.

### Exercice 2

En utilisant les fonctions `sleep( )` et `wait( )`, écrivez un programme dans lequel un processus père crée 10 processus fils et attend qu'ils se terminent. Chaque processus fils attend un nombre de secondes choisi aléatoirement entre 1 et 10, affiche son PID et se termine. A sa terminaison, le processus père affiche le PID du processus fils qui a terminé son exécution.

### Exercice 3

Soient  $m$  et  $n$  deux entiers donnés au lancement d'un programme. Ecrire les programmes décrivant les comportements suivants :

1. Le processus père crée  $m$  fois  $n$  processus fils
2. Le processus père crée  $m$  processus fils qui, chacun à leur tour, crée  $n$  processus fils.
3. Le processus père génère  $n$  processus fils, chacun d'entre eux devant générer  $n$  processus fils à son tour, et ainsi de suite jusqu'à  $m$  niveaux.

Chacun de ces programmes devra calculer et afficher le nombre total de processus générés.

### Exercice 4

Ecrire un programme qui :

1. lance la commande `ls`.
2. redirige la sortie standard de `ls` sur `/dev/null`.
3. affiche, en utilisant la primitive système `times`, le temps pris par la commande `ls`.