

## **TP4**

### **Exercice1 :**

1. A l'aide de la commande "ps", afficher la liste de tous les processus tournant sur la machine. Que signifient les indications présentes ?
2. Lister tous vos processus tournant sur la machine.
3. Est-il possible de déterminer le nombre approximatif de processus tournant actuellement ?
4. Donner la liste des trois derniers "ancêtres" de la commande "ps" en cours d'exécution.
5. Tester la commande "top".

### **Exercice2 :**

1. Créer un programme en c qui incrémente une variable "i" indéfiniment et qui affiche "i" lorsque sa valeur est un multiple de 100. Penser à utiliser la commande "sleep" pour ralentir l'exécution et y voir quelque chose!
2. Compiler le programme avec cette commande : "gcc -o compteur votre\_programme.c".
3. Lancer quatre instances du processus "compteur" en tâche de fond.
4. A l'aide des commandes "jobs", "bg" et "fg", mettre au premier plan la troisième, la suspendre, et la relancer en tâche de fond.
5. Arrêter tous les compteurs avec "ps" et "kill".
6. Tester la commande "killall" pour tuer tous les compteurs.
7. Lancer une instance du processus "compteur" en tâche de fond. Que se passe-t-il si vous fermez le terminal ?
8. Utiliser la commande nohup pour lancer en tâche de fond une instance du processus "compteur" qui continuera à s'exécuter après la fermeture du terminal. Fermer le terminal, vérifier dans un autre terminal qu'elle continue de s'exécuter et arrêter la depuis cet autre terminal.

### **Exercice3 :**

On considère le programme compteur de l'exercice précédent.

1. Écrire un script qui lance ce programme en tâche de fond.
2. Modifier le script afin de lancer en parallèle quatre instances du programme compteur en tâche de fond.
3. Modifier le script afin de lancer les quatre instances de compteur l'une après l'autre et en tâche de fond. Pour cette question, il faut ajouter une valeur limite pour la boucle qui incrémente i. Cette valeur limite doit être suffisamment élevée pour que l'exécution du programme puisse durer plus de 5 secondes.
4. Même question que la précédente sans ajouter de valeur limite pour i, mais en stoppant le processus en cours dès que sa durée d'exécution excède 5 secondes (Utiliser la commande timeout).
5. Modifier le script afin de donner différents niveaux de priorité à ces processus (de moyen à très faible).