

## TD : Les Signaux

### Objectifs :

Utilisation des signaux pour synchroniser des processus

---

### Exercice1 :

Exécutez et analysez le code fourni *pending.c*. vous pouvez envoyer des signaux au processus depuis une autre console et vérifier les résultats obtenus. Vous pouvez par exemple envoyer à la suite les signaux 10, 12, 15. La commande *ps s* vous permet de visualiser les états des signaux.

---

### Exercice2 : Interception d'un signal attendu

Le but de cet exercice est d'écrire un programme qui boucle à l'infini en attendant les signaux SIGUSR1 et SIGUSR2. Lorsqu'il reçoit le signal SIGUSR1, il affiche "*Bonjour, Signal SIGUSR1*" reçu à l'écran et lorsqu'il reçoit le signal SIGUSR2, il affiche "*Bonsoir, Signal SIGUSR2*" reçu.

Le programme doit réagir une seule fois au signal SIGUSR1 en revanche il réagit au signal SIGUSR2 chaque fois qu'il le reçoit.

---

### Exercice3 : Interception d'un signal attendu

Le programme à développer doit permettre :

- ✓ La création de trois processus fils,
- ✓ L'initialisation de deux gestionnaires pour les signaux SIGUSR1 et SIGUSR2,
- ✓ Le fils 1 envoie au père le signal SIGUSR1,
- ✓ Le fils 2 envoie le signal SIGUSR2,
- ✓ Le père se synchronise sur la fin du fils 3

*On pourra utiliser la fonction sleep() pour ralentir le traitement du fils 3.*

---

### Exercice4 : Interception d'un signal et arrêt

Écrire un programme qui affiche un nombre toutes les secondes. Ce nombre sera initialisé à 1 et sera incrémenté à chaque réception du signal d'interruption SIGINT (CTRL C). La réception du signal SIGQUIT arrêtera cet affichage. Le signal SIGTSTP sera ignoré.

---

**Exercice5 : Test d'existence d'un processus**

Créez un programme permettant de "forker" un fils qui exécutera une boucle *while(1)* de sommeil. Une fois le fils créé, le père testera l'existence de son fils et si ce dernier existe, lui enverra un signal SIGUSR1 après une attente de quelques secondes. Le père attendra à nouveau quelques secondes puis se bloquera dans l'attente de la fin de son fils. Lorsque cette fin sera détectée, il cherchera à savoir le type de terminaison et l'éventuelle valeur retournée par le fils (ou le signal ayant terminé ce fils) et les affichera.

*Expliquez l'affichage obtenu. Les attentes de quelques secondes ne sont là que pour avoir le temps d'observer ce qui se passe.*