

TP N°4 : Processus & signaux – 1exercice – Durée approx. ~ 0h30min.

A l'aide du support de cours et des éventuels mémentos réalisez les exercices suivants :

(Tout code ou implémentation compilant ou non sera étudié)

Le présent sujet de TP comporte 2 pages

I – « **Compteur** »

Écrivez un programme comptant le nombre de signal de type **SIGINT** reçu.

- Le processus se compose d'une boucle sans fin qui incrémente un compteur associé au signal **SIGINT** : lorsque l'utilisateur appuie sur **CTRL+C**, le programme affiche le nombre sur la sortie standard.
- A la réception du signal **SIGTERM** le programme affiche un message de fin puis se termine.

IV – Rappels

Approche incrémentale du développement

Pour obtenir les résultats attendus aux différents exercices, veuillez toujours appliquer une approche incrémentale en termes d'ajout de code/fonctionnalité.

Par exemple: une approche incrémentale pour ce type d'exercice "ls-like" serait :

1. La récupération des paramètres.
2. Tester fichier/répertoire.
3. Parcourir les éléments du répertoire en affichant leur nom.
4. Alimenter chaque fichier avec une information supplémentaire : permission / taille / propriétaire...

Documentation

Pour obtenir des informations ou de la documentation ayez le réflexe d'utiliser les pages du manuel.

Par exemple:

- man 3 stat / man 2 open / man 2 readdir / man errno

Gestion des erreurs

Afin d'avoir une gestion des erreurs la plus précise possible ayez le réflexe d'utiliser les codes retours **ERRNO** spécifiés dans les pages de manuel

Par exemple:

- | | |
|------------------|---|
| ➤ EEXIST | File exists (POSIX.1) |
| ➤ EFAULT | Bad address (POSIX.1) |
| ➤ EISDIR | Is a directory (POSIX.1) |
| ➤ ENOTDIR | Not a directory (POSIX.1) |
| ➤ ELOOP | Too many levels of symbolic links (POSIX.1) |