

Signaux

IMA 4 - SC

Exercice 1

Comme échauffement, vous allez réaliser un premier programme qui boucle indéfiniment. Lorsqu'il reçoit le signal `SIGINT`, il affiche un message puis s'arrête. Afin de pouvoir facilement modifier le message sans modifier le code source, vous allez utiliser une macro représentant le message à afficher. Réaliser le Makefile correspondant avec deux cibles afin de produire deux exécutable différents. La première cible sera `perdu` et le message `perdu`. La deuxième cible sera `oups` et le message `oups, encore perdu`.

Exercice 2

Écrire le code d'un processus qui boucle indéfiniment et qui, lorsqu'il est interrompu par `SIGINT` ou `SIGQUIT`, se termine en affichant le signal qui l'a interrompu

Exercice 3

Écrire le code d'un processus qui boucle indéfiniment et qui, lorsqu'il est interrompu par `SIGINT` se suspend pendant 10 secondes, puis reprend le cours normal de son exécution. Faire en sorte que le processus soit ininterrompible par `SIGQUIT` pendant ces 10 secondes. Exécuter le programme et tester pendant les 10 secondes d'ininterrompibilité les signaux `SIGINT` et `SIGQUIT`. Conclure.

Exercice 4

Un processus crée un processus fils. Ce dernier initialise une variable `somme` à 0, puis dans une boucle infini va additionner `somme` à un nombre aléatoire compris entre 0 et 10 puis s'endormir pendant une seconde. Le père s'endort ensuite indéfiniment. Lorsqu'il reçoit le signal `SIGINT`, il tue son fils avant de se terminer lui-même. Le processus fils affiche alors la valeur de `somme` et se termine.

Exercice 5

Reprenez l'exercice précédent. Lorsque le fils reçoit le signal `SIGINT`, il envoie `somme` dans un tube et s'arrête. Le père lit la valeur dans le tube, l'affiche et s'arrête.