

Signaux

Thèmes traités

- Notions de base et mise en œuvre des signaux UNIX.
- Opérations essentielles de l'API signaux Unix, protocole d'usage.
- Programmation d'horloges
- Sauvegarde et restauration de points de reprise (API et protocole d'usage)

1 Questions

1. Qu'apporte le mécanisme d'interruptions à la gestion et la supervision des E/S par le système d'exploitation ? Est-ce que l'utilisation de ce mécanisme est pertinente dans tous les cas ? Pourquoi ?
2. Du point de vue de l'application et du point de vue des mécanismes mis en jeu, quelles sont les différences entre la réception d'un signal et l'appel d'une procédure ?

2 Exercice

Écrire un code C qui imprime le numéro de tout signal reçu (signal émis depuis le terminal ou depuis un autre processus). Ce code indiquera toutes les 3 secondes qu'il est toujours actif et en attente de signaux. Au bout de 5 signaux reçus ou 27 secondes il devra s'arrêter. Le décompte du temps s'appuiera sur la programmation de l'envoi du signal **SIGALRM** par l'horloge de temps physique.

1. dans un premier temps, on utilisera la fonction **alarm** pour l'envoi de **SIGALRM**, et on ne distinguera pas la source des signaux **SIGALRM**.
2. traiter la question précédente en utilisant les timers.
3. les signaux **SIGALRM** peuvent résulter d'appels à **kill**, et non de la programmation de l'horloge. Compléter le code précédent pour réduire (ou éliminer) cette possibilité.

3 Déroulement

- présentation du mécanisme de signaux, mise en œuvre du masque, signaux masqués et pendants (planches 3-5 des transparents d'accompagnement)
- revue rapide des opérations de base (**signal/sigaction**, **kill**, **pause**, **alarm**) (planches 6,8 à 10)
schéma de base de mise en place d'un traitant (planche 7)
Exemple : retour sur l'ordonnanceur, avec le traitant de **SIGALRM**
- exercice 2.5.3 sans timers
- revue rapide des opérations de manipulation des masques de signaux (**sigsuspend/sigprocmask/sigpending**) (planche 10)
- présentation des timers (planche 14). Exercice 2.5.3 avec timers
- présentation de **setjmp/longjmp**.
Illustration avec le principe de mise en œuvre des exceptions (planche 13)