

TP prog sys numéro 4 : IPC

ISIMA ZZ 2

1 Liste des ipc

- recuperez le programme shmtest.c
- lancez le une fois pour voir si il fonctionne
- tapez ipcs
- lancer le programme
- tapez ipcs pendant qu'il fonctionne
- relancez le programme et tapez CONTROL+C quand il affiche un .
- tapez ipcs
- effacez le shm restant à l'aide de ipcrm

2 L'horloge suisse (à quartz d'IPC)

Le but de cette partie est de fabriquer un système de processus apte à gérer un coucou. Dans cette partie, l'utilisation de wait et sleep est prohibée. Les boucles infinies se feront à l'aide de for (; ;) ;

- Faire un programme qui lance trois processus, le premier suis le timer système à l'aide la fonction time, et inscrit les secondes dans un sémaphore, le deuxième suis les secondes et met à jours un sémaphore contenant les minutes, le troisième suis le sémaphore précédent et mes à jours un sémaphore contenant les heures.
- Faire un programme qui affiche l'heure en lisant les semaphores.
- Faire un programme qui se met tout seul en tache de fond qui suis les sémaphore et déclenche un nombre de bip proportionnel a l'heure (utilisé \G)
- Faire un programme kecoui qui arrête l'ensemble et nettoie les sémaphores oubliés.

Remarque : Pour faire vos tests, accélérer le temps et utiliser l'envoi de signaux avec la commande unix kill !

3 La vue !

- Reprogrammer l'exercice de floutage d'image en trois processus gérant respectivement le rouge, le vert et le bleu, à l'aide de trois shared memory.
- Calculer la moyenne et l'écart-type du temps nécessaire à traiter un pixel. Que remarque-t-on ?
- Calculer le taux d'asynchronisme (distance entre les pixels traités par les trois processus). Que remarque-t-on ?

4 L'ouie

- Programmer en utilisant un tableaux de sémaphore et une shared memory une application permettant de s'échanger des messages à trois ou plus.
- Améliorer ce programme pour qu'il permette de passer en mode discution à deux.
- Améliorer ce programme pour qu'il permette de s'envoyer un fichier.