

IPC

IMA 4 - SC

Exercice 1

Comme échauffement, vous allez réaliser un premier programme `pong.c` qui crée une file de messages IPC puis dans une boucle infini, va lire dans la file les messages de type 1 et y répondre par un message de type 2. Lorsqu'il reçoit le signal `SIGINT`, il affiche un message, détruit la file de messages, puis s'arrête. Écrire un second programme, `ping.c`, qui recherche la file de message créée par le premier programme, envoie un message de type 1, attend une réponse de type 2, affiche le temps d'aller-retour, puis s'arrête.

Exercice 2

Écrire un programme `serveur.c` (service de calcul) qui crée une file de messages et attend indéfiniment des messages. Les messages peuvent être de type 10 ou 20. Un message de type 10 correspond à une demande d'addition où les deux nombres sont contenus dans le message. Un message de type 20 correspond à une demande de multiplication où les deux nombres sont contenus dans le message. Chaque opération déclenche le lancement d'un thread addition ou multiplication. Le résultat du calcul est la valeur de retour du thread. Le résultat est renvoyé dans la file de message par un message de type 30. Lorsque ce programme reçoit le signal `SIGINT`, il affiche un message, détruit la file de messages, puis s'arrête. Écrire le programme `client.c` qui demande à l'utilisateur d'entrer les informations (type de calcul, valeurs) puis envoie le message, attend le retour, l'affiche puis se termine.

Exercice 3

En partant de l'exercice précédent, nous voulons maintenant avoir plusieurs clients simultanément. Pour que la réponse arrive au bon demandeur, chaque client va créer une file de messages pour récupérer la réponse. Lors de l'envoi de son message, il indiquera, en plus des deux nombres, la clé de sa file de messages. Côté serveur, cette valeur sera donnée au thread qui pourra ainsi faire directement l'envoi de la réponse.