

R3.05 - TP 2 bis

Signaux (suite)

Chronomètre

Pour les besoins d'une course, vous allez mettre en place un chronomètre qui va compter le nombre de tours effectués par les coureurs, dont on ne connaît pas le nombre à l'avance mais qui ne sont pas plus de 50, et calculer le temps de course total écoulé ainsi que la moyenne sur les trois derniers tours.

Il s'agit d'une course au temps mais les participants entrent en course quand ils le veulent, il n'y a pas de départ unique, avec tous les coureurs en même temps.

Pour ce faire, vous allez écrire deux programmes : **chrono** et **coureur**.

Programme chrono

Il s'agit d'un programme qui boucle sans fin.

Sur réception du signal **SIGUSR1**, ce programme doit :

- Créer un nouveau coureur dans sa "liste". Le numéro du coureur sera son rang dans cette liste.
- Initialiser les informations utiles pour ce coureur. Il doit notamment mémoriser son temps Unix de début de course.

Sur réception du signal **SIGUSR2**, ce programme doit :

- Compter un passage supplémentaire pour le coureur (un tour de plus)
- Calculer ses statistiques : temps de course total, moyenne sur les trois derniers tours.
- Afficher son numéro de coureur et ses statistiques.

Sur réception du signal **SIGTERM**, ce programme doit :

- Lister tous les coureurs et leurs statistiques.
- Quitter.

Vous identifierez un coureur grâce au **PID** de l'émetteur du signal. Sur un **SIGUSR1** vous vérifierez que le coureur n'est pas déjà connu dans la liste et vous mémoriserez son **PID** pour la suite. Sur un **SIGUSR2** vous vérifierez que le coureur est bien connu dans la liste, grâce à son **PID**.

Programme coureur

Il s'agit d'un programme qui tourne en boucle en faisant ces actions :

- Affichage un menu avec 3 choix :
 - 1) Départ
 - 2) Signaler un passage
 - 3) Quitter la course
- Lecture au clavier (**scanf**) un choix entre **1** et **3**
- Traitement de l'action en fonction du choix :
 - 1) Envoie un **SIGUSR1** à **chrono** et retourne au menu
 - 2) Envoie un **SIGUSR2** à **chrono** et retourne au menu
 - 3) **exit()** le programme

Il faudra certainement passer le **PID** du processus **chrono** à chaque lancement du programme **coureur**.

Tests

Lancez le programme **chrono** dans un Terminal.

Identifiez le **PID** de ce processus, soit par un affichage soit par une commande ps.

Dans autant d'autres terminaux que nécessaires (3 mini), lancez le programme **coureur** avec sans doute le **<PID_du_chrono>** en paramètre, et faites des choix dans le menu de chaque coureur en faisant quelques secondes de pause entre chaque tour.

Vérifiez les statistiques qui s'affichent pour chaque passage.

Mettez fin à la course de chaque participant.

Arrêtez chono à l'aide d'un **CTRL+C** et vérifiez le résumé de course qui s'affiche.