Licence Informatique

Université Paul Sabatier
118 route de Narbonne
31062 Toulouse Cedex 09

SR2 – Partie « Systèmes » - Travaux pratiques n°2
Processus Unix – Signaux & Tubes

NB : Deux exercices sont à faire avant la séance 2 – Voir le sujet sous Moodle

Exercice 1 – Envoi périodique d'un message vers un fils

Compléter le programme écrit pour l'exercice 1 de préparation afin que le processus père envoie à son fils le message à intervalles de temps **réguliers** un certain nombre de fois.

Le délai (en secondes) entre deux envois, et le nombre d'envois à réaliser constituent les **paramètres** de cette application.

Le père **peut** continuer son traitement de fond (par exemple, un calcul quelconque) entre les envois de son message (gestion asynchrone) et doit se terminer en dernier.

Il est **interdit** d'utiliser la primitive *sleep()*.

Exemple d'exécution : ./exo1 2 5

/ Un message envoyé 5 fois au fils, toutes les 2 secondes */*

```
Fils - Recu de mon pere : Message numero 0  envoye a  Wed Mar  8 17:06:50 2023
Fils - Recu de mon pere : Message numero 1  envoye a  Wed Mar  8 17:06:52 2023
Fils - Recu de mon pere : Message numero 2  envoye a  Wed Mar  8 17:06:54 2023
Fils - Recu de mon pere : Message numero 3  envoye a  Wed Mar  8 17:06:56 2023
Fils - Recu de mon pere : Message numero 4  envoye a  Wed Mar  8 17:06:58 2023
Fils - Je me termine a Wed Mar  8 17:06:58 2023
Pere - Je me termine en dernier a Wed Mar  8 17:06:58 2023
```

Exercice 2 – Envois périodiques + gestion de la terminaison des fils

Compléter le programme écrit pour l'exercice 2 de préparation afin que les NF capteurs (fils) envoient la valeur de leur capteur au processus père à intervalles de temps **réguliers** (et non plus à chaque fois que NBV véhicules sont captés). Entre deux envois, un capteur doit **continuer** à détecter les véhicules (saisir des caractères).

Pour se terminer, un capteur attendra de **recevoir** l'ordre de son père.

Le père continue à jouer le rôle d'un panneau d'affichage. Lorsque le compteur d'un capteur atteint la valeur NBL, il avertit ce capteur qu'il peut se terminer.

Il se termine lui-même quand tous les capteurs ont fini leur exécution.

Il est **interdit** d'utiliser la primitive *sleep()*.

Exemple d'exécution : ./exo2 2 4 2

/ 2 capteurs, envoyant leur valeur toutes les 2 secondes, à concurrence de 4 captés – On a saisi les caractères abcdeghf pour simuler les véhicules */*

```
abcdefPere (169) - Capteur 1 : nombre de vehicules = 0
Pere (169) - Capteur 0 : nombre de vehicules = 0
        Capteur 0 (170) : Valeur 0 ecrite dans tube a Thu Mar  9 23:30:20 2023
        Capteur 1 (171) : Valeur 0 ecrite dans tube a Thu Mar  9 23:30:20 2023
gh
ijPere (169) - Capteur 0 : nombre de vehicules = 3
        Capteur 0 (170) : Valeur 3 ecrite dans tube a Thu Mar  9 23:30:22 2023
        Capteur 1 (171) : Valeur 6 ecrite dans tube a Thu Mar  9 23:30:22 2023
Pere (169) - Capteur 1 : nombre de vehicules = 6
Pere (169) - Signale au capteur 1 (171) sa fin
        Capteur 1 (171) : Termine

Pere (169) - Capteur 0 : nombre de vehicules = 6
        Capteur 0 (170) : Valeur 6 ecrite dans tube a Thu Mar  9 23:30:24 2023
Pere (169) - Signale au capteur 0 (170) sa fin
        Capteur 0 (170) : Termine
Pere (169) - Je me termine en dernier
```