

COMPTE RENDU - TP2

FEKIH TAOUFIK

1. Fork

L'incrémentation du MAX_COUNT en le multipliant par 10, les printf des fils et des parents ne suivent pas le même ordre qu'en ayant le MAX_COUNT a nombre petit (10 par exemple).

Il paraît que quand on augmente le nombre de processus, le scheduler du système d'exploitation commence à prendre des décisions à propos du temps et de la priorité d'exécution de ces derniers.

```
This line is from child , value = 185
This line is from child , value = 186
This line is from child , value = 187
This line is from child , value = 188
This line is from child , value = 189
This line is from child , value = 190
This line is from child , value = 191
This line is from child , value = 192
This line is from child , value = 193
This line is from child , value = 194
This line is from child , value = 195
This line is from child , value = 196
This line is from child , value = 197
This line is from child , value = 198
This line is from child , value = 199
This line is from child , value = 200
*** Child process is done ***
```

2. Orphelin

On remarque au début que le processus père 8731 et le processus fils 8732 coexistent de façon normale. Après la mort du père, on constate que le fils est encore en vie, maintenu en état de sommeil pendant 40 secondes.

Dans ce cas, le processus fils est nommé “orphelin” car son processus père est mort sans attendre la fin des instructions de ce dernier.

```
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ make
gcc orphelin.c -o main
./main
Je suis le pere (8731)
Je suis le fils (8732) et le PID de mon pere est (8731)
Vous avez 15 sec pour lancer un ps -e -f !

Je suis le fils (8732) et le PID de mon pere est (8731)
Je suis le pere (8731) et je meurs AVANT mon fils (8732)
Fin
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ Je suis donc devenu orphelin !
Fin
```

```
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8731 8732
  PID TTY          STAT       TIME COMMAND
 8731 pts/0        S+          0:00 ./main
 8732 pts/0        S+          0:00 ./main
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8731 8732
  PID TTY          STAT       TIME COMMAND
 8731 pts/0        S+          0:00 ./main
 8732 pts/0        S+          0:00 ./main
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8731 8732
  PID TTY          STAT       TIME COMMAND
 8732 pts/0        S           0:00 ./main
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8731 8732
  PID TTY          STAT       TIME COMMAND
 8732 pts/0        S           0:00 ./main
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8731 8732
  PID TTY          STAT       TIME COMMAND
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$
```

3. Zombie

Un processus nommé “Zombie” est un processus fils qui meurt au moment où le processus père est encore en marche/mise en état de sommeil, et que ce dernier n’a pas encore pris en compte le code de retour du statut du fils.

Le processus fils dit “Zombie” est identifié avec la marque “<defunct>”.

Le processus zombie semble être insensible à la commande “kill -9”. Pour tuer le processus zombie, il faut impérativement tuer le processus père.

```
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ rm main
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ make
gcc zombie.c -o main
./main

Je suis le pere : PID 8237
Qu'est devenu mon fils 8238
Vous avez 30 sec pour lancer un pis -e -f et constater qu'il est zombi !
Je suis le fils : PID 8238 et le PID de mon pere est 8237
Je suis le fils et je meurs : PID 8238

Makefile:4: recipe for target 'main' failed
make: *** [main] Killed
make: *** Deleting file 'main'
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ █
```

```
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8237 8238
  PID TTY          STAT       TIME COMMAND
 8237 pts/0      S+          0:00 ./main
 8238 pts/0      Z+          0:00 [main] <defunct>
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ kill -9 8238
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8237 8238
  PID TTY          STAT       TIME COMMAND
 8237 pts/0      S+          0:00 ./main
 8238 pts/0      Z+          0:00 [main] <defunct>
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ kill -9 8237
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ ps 8237 823
  PID TTY          STAT       TIME COMMAND
elementary@elementary:~/Documents/git/L3-Miage-depot-TP$ █
```

4. Wait

On illustre dans cette partie la façon dont le père se met en mode attente pour récupérer le code de retour de N fils afin d’afficher un message personnalisé sur la console et ainsi éviter la création de processus orphelin/zombie.


```

elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ make
gcc wait.c -o main
./main
Le nombre de fils a creer : 5
Pid du pere = 27485
Attente de la terminaison du fils...
Je suis le fils 27488 et mon pere est 27485
Je suis le fils 27490 et mon pere est 27485
Je suis le fils 27489 et mon pere est 27485
Je suis le fils 27487 et mon pere est 27485
Je suis le fils 27486 et mon pere est 27485
Le fils 27488 s'est terminé correctement : 0
Pid du pere = 27485
Attente de la terminaison du fils...
Le fils 27490 s'est terminé correctement : 0
Pid du pere = 27485
Attente de la terminaison du fils...
Le fils 27489 s'est terminé correctement : 0
Pid du pere = 27485
Attente de la terminaison du fils...
Le fils 27487 s'est terminé correctement : 0
Pid du pere = 27485
Attente de la terminaison du fils...
Le fils 27486 s'est terminé correctement : 0
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ █

```

5. SIGCHLD

SIGCHLD est le signal envoyé par un processus qui vient de mourir à son processus père afin de gérer de façon asynchrone (pas comme la primitive wait qui est bloquante) cet événement et exécuter des instruction d'anticipation.

```

elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ make
gcc sigchld.c -o main
./main
Pid du fils = 1259

Pid du pere = 1257
Pid du fils = 1260
Attente de la terminaison du fils...
Le fils 1259 s'est termine correctement : 0
Attente de la terminaison du fils...
Le fils 1260 s'est termine correctement : 0
Mort du pere!
elementary@elementary:~/Documents/git/L3-Miage-depot-TP/S&R/TP2$ █

```