

Processus

François Trahay



CSC4103 – Programmation système
2019–2020

1 Caractéristiques d'un processus

■ **PID** (*Process IDentifier*) : identifiant unique du processus

◆ `pid_t getpid()` ;

◆ retourne le PID du processus courant

■ **PPID** (*Parent PID*) : identifiant du processus père

◆ `pid_t getppid()` ;

◆ retourne le PPID du processus courant

2 Création de processus

- `int system(const char* cmd);`
- Crée un processus shell qui exécute `cmd`
- Retourne le code de retour de la commande (0 si tout s'est bien passé)

2.1 fork

- `pid_t fork();`
- duplique le processus courant
 - ◆ duplication de la mémoire, des fichiers ouverts, etc.
- le processus créé est le fils du processus courant
- le processus fils est une copie exacte du père sauf
 - ◆ le PID du fils est différent
 - ◆ le PPID du fils est le PID du père
- seule la valeur retournée par `fork` permet de différencier le père et le fils
 - ◆ le père reçoit le PID du fils
 - ◆ le fils reçoit 0

2.2 La classe de fonctions exec

- ensemble de fonctions permettant d'exécuter une commande
 - ◆ `exec1p`, `execvp`, `execve`, `execle`, `exec1p`, etc.
- remplace le programme appelant par un nouveau
 - ◆ donc, on ne “sort” jamais de la fonction (sauf erreur)

2.3 Terminaison d'un processus

- `pid_t wait(int *status);`
- Attend la terminaison d'un processus fils
- Le champs `status` permet de connaître la cause du décès.
- Variante: `pid_t waitpid(pid_t pid, int *wstatus, int options);`