PP dp

NCours système

# PID

PID : Process ID

PPID : Parent Process ID

Pid 0 est idle

Pid 1 est le premier process exécuté

# Fonctions :

#include <sys/types.h>

#include <unistd.h>

pid\_t getpid (void); /\*pid\*/

pid\_t getppid (void); /\*parent pid\*/

Affichage des valeurs :

printf ("Mon PID=%jd\n", (intmax\_t) getpid ()); /\*notez le casting\*/

printf ("PID du Parent =%jd\n", (intmax\_t)getppid ()); /\*notez le casting\*/

# Pere-Fils

Un processus qui engendre est le père, celui engendré est le fils.

Arborescence depuis init vers les processus fils

Un pus fils hérite de l’uid (user id) et gid (group id) de son père.

# Nouveau pus : exec et fork

ex

# Terminer pus : exit, atexit

## exit

– appel des triggers

– écriture des données I/O

– destruction des fichiers temporaires

– libération des ressources allouées par le noyau:

mémoire, fichiers ouverts, sémaphores.

– notification au parent de la fin de son processus

fils

#include <stdlib.h>

void exit (int status);

## atexit

#include <stdlib.h>

int atexit (void (\*function)(void));

#include <stdio.h>

#include <stdlib.h>

void out (void)

{

printf ("atexit() fonctionne bien!\n");

}

int main (void)

{

if (atexit (out)) fprintf(stderr, "atexit()

erreur!\n");

return 0;

}