

Test n°2

Processus, signaux, ordonnancement, mémoire, disques

CORRECTION**Exercice 1 – Processus*****Solution :***

```
pid 15557 : 2
pid 15558 : 4
pid 15559 : 8
pid 15560 : 2
pid 15561 : 4
pid 15562 : 4
pid 15563 : 2
```

Exercice 2 – Signaux***Solution :***

- 1) Ligne 3 : il y a création d'un processus fils. En cas de succès, le PID du fils est renvoyé au processus parent, et 0 est renvoyé au processus fils. Aussi, c'est le processus père qui effectue le wait(NULL) et qui attend donc la terminaison de son fils avant de continuer.
Ligne 5 : les 2 processus (père et fils) affichent leur n° de PID suivi de « fini ».

2)

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>

static int drapeau = 0 ;

static void interruption (int signum)
{
    drapeau = 1;
    return ;
}

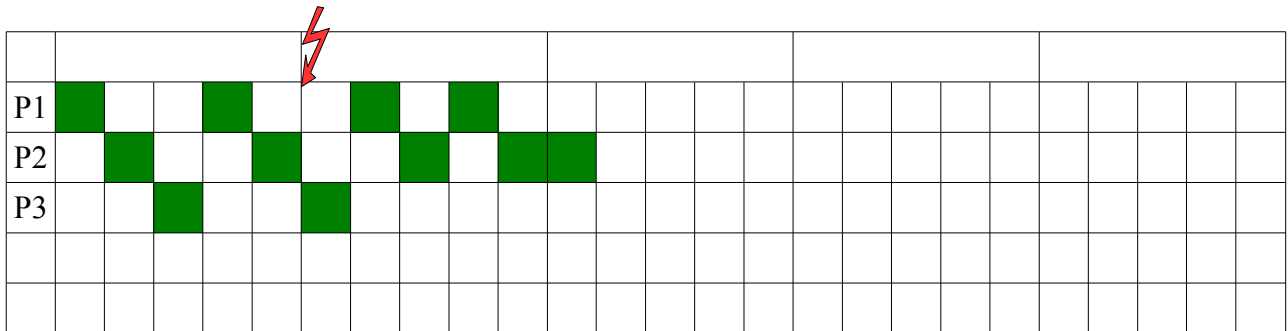
static void (* old)(int signum);

/*Installation du handler sur le signal SIGCHLD*/
void init_wait_simple (void)
{
    old = signal(SIGCHLD , interruption);
}

/*Désinstallation du handler*/
void end_wait_simple (void)
{
    signal (SIGCHLD, old);
}
```

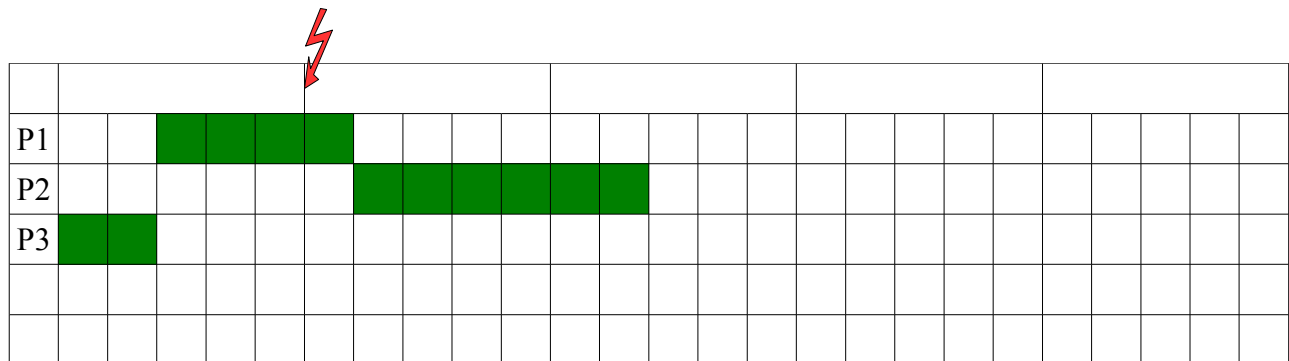
Solution :

Round-Robin (q=1)



P1 ne respecte pas son échéance.

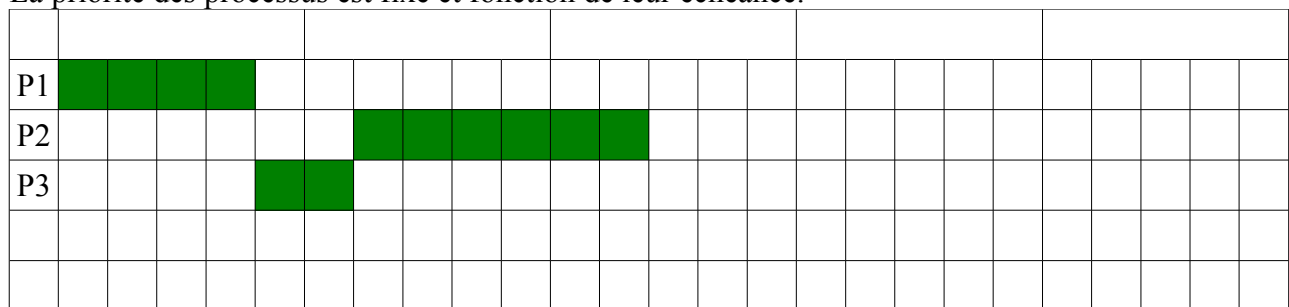
SJF



P1 ne respecte pas son échéance.

Ordonnancement à priorités

La priorité des processus est fixe et fonction de leur échéance.



Tous les processus respectent leur échéance

Exercice 4 – Mémoire

Solution :

1)

L'algorithme OPT (OPTimal) consiste à choisir comme victime la page qui sera référencée le plus tardivement dans le futur

Page	2	1	0	7	1	6	1	5	7	6	2	6	7	0	7	1	0	2	1	7
Cadre 0	2	2	2	2	2	6	6	6	6	6	6	6	6	0	0	0	0	0	1	1
Cadre 1	-	1	1	1	1	1	1	5	5	5	2	2	2	2	2	2	2	2	2	2
Cadre 2	-	-	0	7	7	7	7	7	7	7	7	7	7	7	7	1	7	7	7	7
Déf. de page	0	0	0	0	N	O	N	O	N	N	O	N	N	O	N	O	N	N	N	N

2)

L'algorithme FIFO (First In First Out) consiste à choisir comme victime la page la plus ancienne en mémoire

Page	2	1	0	7	1	6	1	5	7	6	2	6	7	0	7	1	0	2	1	7
Cadre 0	2	2	2	7	7	7	7	5	5	5	2	2	2	2	2	1	1	1	1	1
Cadre 1	-	1	1	1	1	6	6	6	7	7	7	7	7	0	0	0	0	2	2	2
Cadre 2	-	-	0	0	0	0	1	1	1	6	6	6	6	6	7	7	7	7	7	7
Déf. de page	0	0	0	0	N	0	0	0	0	0	0	N	N	0	0	0	N	0	N	N

3)

L'algorithme LRU (Least Recently Used/la moins récemment utilisée) consiste à choisir comme victime la page qui n'a pas été référencée depuis le plus longtemps.

Page	2	1	0	7	1	6	1	5	7	6	2	6	7	0	7	1	0	2	1	7
Cadre 0	2	2	2	7	7	7	7	5	5	5	2	2	2	0	0	0	0	0	0	7
Cadre 1	-	1	1	1	1	1	1	1	6	6	6	6	6	6	1	1	1	1	1	1
Cadre 2	-	-	0	0	0	6	6	6	7	7	7	7	7	7	7	7	7	2	2	2
Déf. de page	0	0	0	0	N	0	N	0	0	0	0	N	N	0	N	0	N	0	N	0

Exercice 5 – Disques

Solution :

1)

FIFO

Positions consécutives de la tête de lecture/écriture :

15 - 100 - 30 - 27 - 55 - 16 - 122 - 44 - 63 - 56.

Le déplacement total est donc : $85 + 70 + 3 + 28 + 39 + 106 + 78 + 19 + 7 = 435$.

2)

SSTF

Positions consécutives de la tête de lecture/écriture :

15 - 16 - 27 - 30 - 44 - 55 - 56 - 63 - 100 - 122.

Le déplacement total est donc : $1 + 11 + 3 + 14 + 11 + 1 + 7 + 37 + 22 = 107$.