



Département d'Informatique
LICENCE INFORMATIQUE 2ème année

Programmation Système TD n° 4

Notions abordées :

- **Signaux POSIX** (`sigaction()`, `sig*set()`, `sigsuspend()`, `alarm()`, `kill()`, `sigprocmask()`)

Exercice 1

Écrire un programme qui boucle sur l'affichage de "bla bla ..." et qui affiche "raté!" et s'endort 1 seconde à chaque fois qu'il reçoit le signal **SIGINT** (Ctrl+C). Comment peut-on arrêter l'exécution de ce programme ?

Exercice 2

Écrire un programme C qui compte les signaux qu'il reçoit et affiche ces compteurs. Exemple d'exécution :

```
$ ./a.out &
[2] 4640
$ kill -USR1 4640
Signal 30 recu 1 fois
$ kill -USR1 4640
Signal 30 recu 2 fois
$ kill -USR1 4640
Signal 30 recu 3 fois
$ kill -CONT 4640
Signal 19 recu 1 fois
$ kill -STOP 4640
[2]+ Stopped ./a.out
$ kill -CONT 4640
Signal 19 recu 2 fois
$ kill -KILL 4640
[2]+ Killed ./a.out
$
```

Exercice 3

Écrire un programme C qui crée un processus fils : Le père affiche la suite des entiers impairs. Le fils affiche la suite des entiers pairs. On doit synchroniser les processus à l'aide des signaux pour que les entiers s'affichent dans l'ordre. Exemple d'exécution :

```
$ ./a.out
Père: 1
Fils: 2
Père: 3
Fils: 4
...
```

Exercice 4

Modifier le programme précédent pour qu'il se termine automatiquement au bout de 10 secondes.