

# 1 Documentation

## 1.1 Signaux sur unix :

Les signaux varient selon quel unix on utilise (à part quelques uns, les plus couramment utilisés). On utilise donc généralement les macros correspondants, ou bien “kill -l” pour connaître la liste .

Liste des 31 premiers <http://www.ucs.cam.ac.uk/docs/course-notes/unix-courses/Building/files/signals.pdf>. Les numéros sont utilisés pour le temps réel, et ne nous concernent pas vraiment.

Attention, les signaux sont vulnérables aux race conditions.

Exemple de handling et plus d’informations

## 1.2 Pipes

Petits tutos pour les pipes :

<https://www.cs.rutgers.edu/~pxk/416/notes/c-tutorials/pipe.html>

Envoyer du data formaté par des pipes :

Les fonctions strtol et sprintf pour envoyer tout en tant que des string.

<http://stackoverflow.com/questions/3457487/sending-structured-data-over-named-pipe-linux>

<http://stackoverflow.com/questions/5237041/how-to-send-integer-with-pipe-between-two-processes>

# 2 Indices pour le tp

Plutôt une liste de questions à se poser :

- Que se passe-t-il si plusieurs signaux sont envoyés avant que le récepteur ne prenne en charge l’un d’entre eux ?
- Dans la 2.2, comment faire pour que les processus fils commencent à tirer en même temps ?
- Est-ce que les premiers enfants ont un avantage? Comment le contrer ?
- Comment savoir s’il ne reste qu’un processus vivant? Peut-il en rester 0 ?