

# Tubes et redirections

## ■ Les descripteurs

- L'entrée standard identifiée par le descripteur 0 (clavier)
- La sortie standard par le descripteur 1 (écran ou interpréteur de commande)
- La sortie d'erreur par le descripteur 2 (par exemple l'écran)

■ E.g : `echo 'salut'` redirige le message vers l'écran

■ Changement de sortie, d'entrée ou de sortie d'erreur

- `>` , `echo salut > file`
- `>>`, `echo sale >> file`
- `<`, `commande < chemin : cat < file`
- `cat /etc/passwd 2> fichier_erreur`: redirige la sortie d'erreur vers un fichier
- `>&` redirige la sortie erreur et la sortie standard vers un fichier
- `tail /etc/passwd >& fichier_sortie` ou `tail /etc/passwd fichier_sortie 2>&1`

# Les tubes, tee, caractères spéciaux

- ❑ Tubes = pipes, la sortie d'une commande devient l'entrée d'une autre.  
Cmd 1 | Cmd 2
  - ❑ E.g: `sort /etc/passwd | head -6`
- ❑ Tee: duplique le flux de données en sortie: elle copie la sortie dans un fichier et en même temps affiche le résultat sur la sortie standard et permet de le renvoyer à une autre commande
  - ❑ exemple: `ls -l /etc | tee fichier | wc -l`
- ❑ ; permet d'enchaîner plusieurs commandes sur une même ligne. L'exécution est séquentielle. `mkdir file; cd file`
- ❑ & permet de lancer une commande en tâche de fond (background)  
exemple: `mkdir file1 &`
- ❑ && placé entre deux commandes permet d'exécuter la deuxième commande ssi la 1ere s'est exécutée avec succès.
- ❑ `Cmd 1 || cmd 2` exécute la 2e ssi la 1ere renvoie une erreur

# Variables d'environnement

- ❑ Variables simples sont modifiés avec “=”
- ❑ L'accès au contenu de la variable se fait en le précédant de \$.
- ❑ L'affichage se fait avec *echo*. *echo \$PATH*.
- ❑ *Pour avoir toutes les variables d'environ, taper env*
- ❑ *unset variable* détruit la variable.

# Les processus

- ❑ Généalogie: *ps tree -pu*
- ❑ *ps aux*, affiche la liste des processus courants
- ❑ *top*: donne les informations sur l'activité sur système
- ❑ *Kill -num\_signal pid* : permet d'envoyer un signal au processus (num\_signal cf. cours lpic1, page 74)
- ❑ *Kill -9 pid ou killall nom\_processus*: tue le processus
- ❑ *nice* : permet de modifier la priorité d'un processus.
- ❑ *Nohup*: lance le processus qui pourra rester aussi longtemps que nécessaire
- ❑ *fg pid ou bg pid* : faire passer un proc en background ou en foreground
- ❑ *Utilisation de &* : équivaut à bg