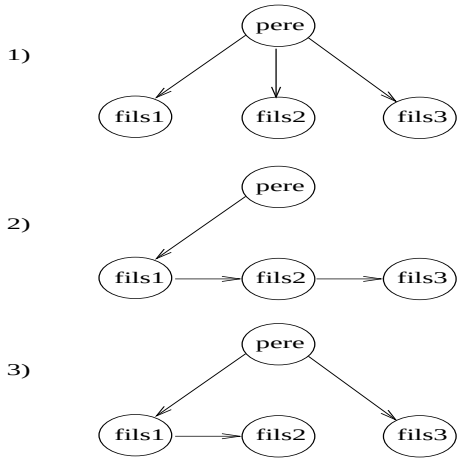
Programmation Système & Réseau
TP noté (1h15)

Exercice 1 - Famille et descendance (20')

Écrire un programme C qui engendre 3 processus des 3 façons différentes représentées sur la figure ci-contre. Chacun des 3 processus doit afficher son pid ainsi que celui de son père (cf. exemple d'exécution). Chaque processus père doit attendre la terminaison de ses fils avant de terminer.



Exemple d'exécution :

```
$./a.out
1)
Je suis fils2 mon numero est le : 22720, mon pere est le : 22718
Je suis fils3 mon numero est le : 22721, mon pere est le : 22718
Je suis fils1 mon numero est le : 22719, mon pere est le : 22718
2)
Je suis fils1 mon numero est le : 22722, mon pere est le : 22718
Je suis fils2 mon numero est le : 22723, mon pere est le : 22722
Je suis fils3 mon numero est le : 22724, mon pere est le : 22723
3)
Je suis fils1 mon numero est le : 22725, mon pere est le : 22718
Je suis fils3 mon numero est le : 22726, mon pere est le : 22718
Je suis fils2 mon numero est le : 22727, mon pere est le : 22725
$
```

Exercice 2 - Des chiffres et des lettres (55')

Écrire un programme C qui implémente l'application suivante : Un processus père lit des caractères sur l'entrée standard et filtre pour l'un de ses fils les lettres et pour l'autre les chiffres (On utilisera les fonctions `isalpha(c)` et `isdigit(c)`.). Le premier fils compte le nombre de lettres pendant que le second compte le nombre de chiffres. Lorsque le père atteint la fin de fichier sur l'entrée standard, il en avise ses fils qui envoient leurs résultats au père qui se charge de les afficher sur la sortie standard. Père et fils communiquent via des tubes de communication.

Exemple d'exécution :

```
$./a.out
dfj;/:567fg
^D
nb de chiffres = 3
nb de lettres = 5
$
```