

## TD ASR4

### PIPE et REDIRECTION

Réalisez un programme C permettant de créer les ressources nécessaires à l'exécution de commandes sur le modèle suivant :

```
$ commande1 -options1 argument1 | commande2 -options2 argument2 >file
```

Pour cela il faut :

- s'assurer que le nombre de paramètres requis est bien fourni :

```
if (argc < 8) { printf(" usage :%s cmd1 -opt1 arg1 cmd2 -opt2 arg2 file ") ; exit(1); }
```

- créer un tube grâce à l'appel système pipe :

```
int p[2 ] ;  
if(pipe(p)){ perror(" pipe ") ; exit(1); }
```

- créer un processus fils pour exécuter la commande2 par l'appel système fork() :

```
switch(fork()){ case ... }
```

- dans le processus fils créer le fichier *file* par l'appel système open

```
int fd ;  
if ((fd=open(name, mode, perm))==-1) { perror("open") ; exit(2); } }
```

- pour réaliser les redirections d'E/S on utilise les fonctions suivantes:

```
sortie vers le tube: close(1) ; dup(p[1]) ;  
sortie vers le fichier : close(1) ; dup(fd) ;  
entrée depuis le tube : close(0) ; dup(p[0]) ;
```

- dans chacun des processus père et fils lancer l'exécution des commandes1 et commande2 par l'appel système :

```
execlp( cmd, cmd, opt, arg, NULL) ;
```

#### Exemple d'utilisation :

```
$ ./prog ls -l /etc/ grep -e "m" file
```