Ressource R3.05 - TD 6

sf **Tubes, redirections**

1. Écrire un programme qui fait la même chose que le bash avec

```
ls | wc >> output.txt
```

2. Expliquez précisément le fonctionnement du programme suivant :

```c
// include
void pere(int fdin){
  char c;
  while(1){
    ssize_t n=read(fdin,&c,1);
    if (n<=0) break;
    if (islower(c)) c-=32;
    write(STDOUT_FILENO,&c,1);
  }
  close(fdin);
}
void fils(int fdout){
  char c;
  while(1){
    ssize_t n=read(STDIN_FILENO,&c,1);
    if (n<=0) break;
    write(fdout,&c,1);
  }
  close(fdout);
  exit(0);
}
int main (int argc,char * argv[]){
  pid_t pid;
  int p[2];
  assert(pipe(p) == 0);

  pid=fork();
  assert(pid >=0);

  if (pid == 0) {
    close(p[0]);
    fils(p[1]);
```

```
33    }else {
34      close(p[1]);
35      pere(p[0]);
36    }
37    wait(NULL);
38    exit(0);
39  }
```

3. On se propose de calculer les nombres premiers inférieurs ou égaux à une valeur `N`, passée à la ligne de commande. Pour ce faire, on utilise le crible d'Ératosthène. Il s'agit de mettre en place la situation suivante :

```
            ------
2,...,N -> ()    () --> P
            ------

P reçoit 2, qui est donc premier, et crée F2

            ------             ------
2,...,N -> ()    () --> F2 -->()    () -> P
            ------             ------

F2 filtre tous les multiples de 2.
P reçoit 3, qui est donc premier, et crée F3

            ------             ------             ------
2,...,N -> ()    () --> F2 -->()    () -> F3 -> ()    () -> P
            ------             ------             ------
Ainsi de suite
```

   (a) Le processus initiale P crée un fils relié par un tube qui y écrit tous les entiers de $2, \ldots, N$.

   (b) Quand un entier parvient jusqu'à P, il est premier. P crée un nouveau filtre.