

# Taller de IPC

## Sistemas Operativos

Primer cuatrimestre de 2020

### Ejercicio: Mini *shell*

En este taller, realizaremos la implementación de un *shell* minimal. El mismo soportará ejecutar programas con sus argumentos (por ejemplo, `ls -al`), permitiendo además comunicar dos o más de ellos separándolos con `|` (por ejemplo, `ls -al | wc | awk '{ print $2 }'`), de modo que la salida de cada programa de la lista sea tomada como entrada por el programa siguiente. Esta forma de ejecutar programas se conoce como *pipeline* y es habitual en cualquier *shell* Unix.

La cátedra les brinda un esqueleto que simula la ejecución del comando

```
ls -al | wc | awk '{ print $2 }'.1
```

Deberán completar el código de forma que el comportamiento sea el mismo que se obtendría al ejecutar dicho comando en cualquier otro *shell*. No se puede utilizar la función `system` para dicha tarea.

Algunas cosas a tener en cuenta:

- En un *pipeline*, todos los programas se ejecutan al mismo tiempo (de manera **concurrente**).
- La comunicación entre los programas se realiza a través de *pipes*.
- Es importante no dejar abiertos los *pipes* que no se vayan a utilizar. En este caso, ¿qué sucede si no se cierran los *pipes*? ¿Por qué?
- En el material de la clase práctica sobre IPC podrán encontrar información útil y algunos ejemplos de código que pueden servirles. También podrían ser de utilidad las páginas del manual `man 7 pipe`, `man 2 pipe` y `man 2 dup`.

---

<sup>1</sup>Si lo desean, pueden modificar la función `main` para probar la ejecución de diferentes comandos.