

## Contenido:

- `fork()`
- `execl()`
- `getpid()`
- `getppid()`
- `close()`
- `open()`
- `perror()`
- `exit()`
- `snprintf()`
- `rand()`
- `srand()`

## Enunciado:

Se deberá crear una jerarquía de procesos como la que muestra la figura:

Donde el proceso A deberá hacer lo siguiente:

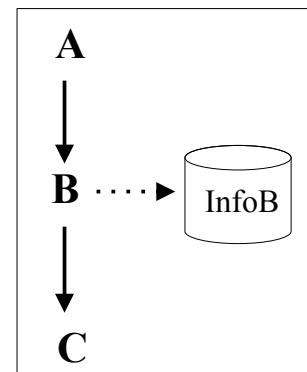
- Muestra un mensaje de bienvenida.
- Lanza el proceso B, redireccionando su salida estándar hacia un fichero llamado *InfoB*.
- Muestra su PID y el PID de B.
- Espera dos segundos antes de terminar.

El proceso B deberá hacer lo siguiente:

- Muestra un mensaje de bienvenida.
- Genera un número aleatorio entre 1 y 10.
- Lanza el proceso C, pasándole como argumento el número aleatorio generado.
- Muestra su PID, el PID de A y el PID de C.
- Espera un segundo antes de terminar

El proceso B, ejecutado por separado deberá mostrar información por la pantalla, pero cuando lo crea el proceso A, la información deberá guardarse en el fichero '*infoB*'.

El proceso C mostrará un mensaje de bienvenida con su PID, el PID de B y el número que se le pasa como parámetro.



### NOTA:

Tened en cuenta que la tabla de canales de B está modificada al haber sido redireccionada la salida estándar al fichero *infoB*, tras ser creado por A. Por tanto, para que C pueda visualizar información por pantalla el proceso B debe pasarle a C una tabla de canales que contenga la salida estándar. En otro caso, C también escribiría en *InfoB*, en lugar de la pantalla.