

## Criterios de evaluación de la práctica 1

**José Carlos Torró Belda**

Las tablas que aparecen a continuación muestran los criterios de evaluación que seguiré para corregir los dos ejercicios del boletín de la práctica 1 que tenéis que entregar. Antes de dejar en el Aula Virtual el código de dichos ejercicios, revisad si la solución que habéis dado para estos ejercicios se ajusta a dichos criterios de evaluación. Estas tablas os permitirán supervisar y mejorar la calidad de vuestro trabajo antes de entregarlo. Y **recordad que es imprescindible dejar en el Aula Virtual las tablas con los criterios de evaluación rellenadas para que os evalúe los ejercicios que habéis entregado.**

<b>Ejercicio 8</b>	<b>Sí</b>	<b>No</b>
1. Si no existe el fichero pasado como primer argumento, se crea el fichero y se vuelca en él la salida del comando o programa ejecutado	X	
2. Si existe el fichero pasado como primer argumento, la salida del comando ejecutado se vuelca al final del contenido original de dicho fichero	x	
3. Si no existe el fichero pasado como segundo argumento, se crea el fichero y se vuelca en él la salida de error del comando ejecutado	x	
4. Si existe el fichero pasado como segundo argumento, se sobrescribe el contenido del fichero con la salida de error del comando	x	
5. Los ficheros que se crean en el programa para guardar la salida y la salida de error, tienen los permisos adecuados	x	
6. El proceso hijo ejecuta el programa o comando especificado a partir del tercer argumento	x	
7. El programa tiene en cuenta que puede fallar la ejecución de la función <code>execvp</code> , muestra el mensaje de error generado y actúa en consecuencia	x	
8. El código de terminación del programa es el código de terminación del proceso hijo	x	
9. ¿Tiene el programa errores o avisos ("warnings") de compilación?		x
10. ¿Tiene el programa errores de ejecución?		x
Aspectos adicionales que quieras hacer constar sobre la autoevaluación del ejercicio:		

**SOFTWARE DE SISTEMAS Y DE COMUNICACIONES (EI1058)**

Grado en Ingeniería Informática

<b>Ejercicio 12</b>	<b>Sí</b>	<b>No</b>
1. Se sigue un esquema de sincronización entre procesos similar a los empleados en los ejercicios del EA1	X	
2. Se justifica cuáles de los esquemas de sincronización entre procesos del EA1 pueden ser válidos y cuáles no, así como la razón del esquema elegido	x	
3. El programa crea tantos procesos como tablas menos uno	x	
4. Cada proceso imprime su PID y su tabla	x	
5. En primer lugar se imprime la tabla del número 1, a continuación la del 2 y así sucesivamente		x
6. No se mezcla la salida de las diferentes tablas y queda claro qué proceso imprime cada tabla	x	
7. Si se incluye una función <code>sleep(3)</code> inmediatamente antes del bucle que imprime una tabla, ¿se mezcla la salida de las diferentes tablas?		x
8. El programa crea tantas tuberías como tablas menos uno	x	
9. En el código del programa se define una única variable de tipo tubería para crear todas las tuberías que se utilizan	x	
10. ¿Bloquea algún proceso la creación de otro proceso?		x
11. ¿Hay espera activa en la sincronización entre procesos?		x
12. El prompt aparece cuando se han impreso todas las tablas	x	
13. ¿Aparecen funciones <code>wait</code> de forma innecesaria (bloqueando, por ejemplo, la ejecución concurrente de los procesos de forma innecesaria)?		x
14. ¿Tiene el programa errores o avisos (“warnings”) de compilación?		x
15. ¿Tiene el programa errores de ejecución?		x
Aspectos adicionales que quieras hacer constar sobre la autoevaluación del ejercicio:		

## **Justificación**

En el ejercicio 12 he utilizado el esquema de ABUELO – PADRE – HIJO porque respecto a los otros esquemas, el redireccionamiento es más sencillo; es decir, el padre puede directamente escribir en la tubería sin necesidad de redireccionar la salida estandar, además, esta salida la necesita para poder imprimir por pantalla la multiplicación.

Así mismo, para poder sincronizar los procesos usamos las funciones `read()` y `write()` con ello hacemos que la salida por pantalla sea ordenada. También usamos una función `wait(NULL)`; para hacer que el prompt salga cuando termine la ejecución de todo el programa.

En conclusión, elegimos este esquema por facilidad de redireccionamientos ya que no resultan ningún problema para poder escribir en la tubería y mostrar por pantalla.