**MEMORIA PRACTICA 2 SISTEMAS OPERATIVOS 2016-2017**

**Ejercicio 2:**En este ejercicio declaramos una memoria compartida entre n procesos. En primer lugar, creamos todos los procesos con un bucle y comprobamos que se crean todos.

Posteriormente realizamos la escritura y lectura en terminal y memoria compartida.

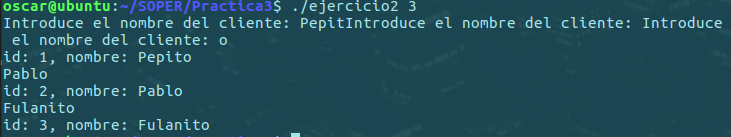
Para el padre realizamos un captador de señales personalizado vacío para poder situar un pause(). Leemos el nombre y el id del buffer y lo imprimimos.

Para los hijos, abrimos la memoria compartida y esperamos un tiempo aleatorio, para conseguirlo y evitar igualdades utilizamos como semilla el id del proceso. Leemos el cliente y aumentamos el id de la estructura. También declaramos un máximo de tiempo de espera para no eternizarnos. Al finalizar la ejecución esperamos a todos los hijos a que terminen si no lo han hecho ya y liberamos la memoria compartida.

El error de este programa es no tener en cuenta la concurrencia en un mismo recurso de varios procesos:

La terminal queda compartida, lo que hace que no sepas a que proceso le estas enviando el nombre de usuario

La memoria compartida no respeta turnos de actuación, lo que hace que puedan no concordar id con nombre de usuario.



**Ejercicio 4:**

En este ejercicio implementamos una librería para la gestión de semáforos.

Hemos realizado todas las funciones de la manera más correcta posible, con control de errores y impresiones por pantalla en caso de error.

**Ejercicio 5:**

En este ejercicio hemos realizado los tests para la biblioteca semáforos.

En caso de ser ejecutado sin argumentos ejecutará todos los tests en orden, pero si se le llama con un argumento de entrada (un número) ejecutará ese test en concreto, permitiendo la individualidad de tests.

Una vez finalizados todos los tests vemos que nuestra biblioteca es válida: