

CC4302 Sistemas Operativos

Tarea 3 – Semestre Primavera 2019 – Prof.: Luis Mateu

En esta tarea Ud. deberá implementar un driver para Linux que permita formar moléculas de agua. Una molécula de agua se forma de 2 átomos de hidrógeno y uno de oxígeno. Cada átomo de hidrógeno se suministra mediante una escritura en el dispositivo `/dev/h2o` y un átomo de oxígeno se suministra con una lectura de ese mismo dispositivo. La lectura o escritura que suministra un átomo debe esperar hasta que se forme la molécula completa. La única excepción es que una lectura puede ser interrumpida con *control-C*. El resultado de la lectura (átomo de oxígeno) es la concatenación de lo escrito al suministrar los átomos de hidrógeno. El dispositivo `/dev/h2o` debe tener número mayor 60. Los átomos de hidrógeno se asocian con los de oxígeno en orden FIFO.

El siguiente ejemplo usa los comandos estándares de Unix *echo* y *cat* para demostrar el comportamiento esperado de `/dev/h2o`. Su driver debe reproducir exactamente el mismo comportamiento. Si hay aspectos que el ejemplo no aclara, decida Ud. mismo tratando de simplificar su tarea. Las filas de la tabla están ordenadas cronológicamente. Lo que escribió el usuario aparece en **negritas**.

Shell 1	Shell 2	Shell 3	Shell 4
\$ echo abc > /dev/h2o ⁽¹⁾			
	\$ echo def > /dev/h2o ⁽¹⁾		
		\$ echo ghi > /dev/h2o ⁽¹⁾	
\$ ⁽³⁾	\$ ⁽³⁾		\$ cat < /dev/h2o ⁽¹⁾ abc def ⁽³⁾
	\$ echo jkl > /dev/h2o ⁽⁴⁾ \$ ⁽⁵⁾	\$ ⁽⁵⁾	ghi jkl
\$ echo mno > /dev/h2o ⁽⁶⁾			
\$ ⁽⁸⁾		\$ echo pqr > /dev/h2o ⁽⁷⁾ \$ ⁽⁸⁾	mno pqr ⁽⁸⁾
			<control-C> \$ ⁽⁹⁾

Notas:

- (1) El comando *echo* se implementa invocando *write* de sus argumentos en la salida estándar. Las 3 escrituras en *shell 1*, *shell 2* y *shell 3* aportan 3 átomos de hidrógeno y se bloquean a la espera de formar moléculas de agua.
- (2) El comando *cat* en *shell 4* invoca *read* de su entrada estándar. Esto aporta un átomo de oxígeno.
- (3) Se forma la primera molécula de agua. Los comandos *echo* en *shell 1* y *shell 2* terminan mientras que *cat* en *shell 4* muestra en su salida estándar el texto aportado por ambos shells. En seguida *cat* realiza una segunda lectura que aporta un segundo átomo de oxígeno, pero como hay un solo átomo de hidrógeno, tanto *cat* en *shell 4* como *echo* en *shell 3* continúan esperando.
- (4) Se aporta un cuarto átomo de hidrógeno en *shell 2*.
- (5) Se forma la segunda molécula de agua. Los comandos *echo* en *shell 2* y *shell 3* terminan y *cat* muestra el texto aportado por estos 2 *echo*. El comando *cat* realiza una tercera lectura (3^{er} átomo de oxígeno) y se queda esperando la llegada de 2 átomos de hidrógeno.
- (6) *Shell 1* aporta un quinto átomo de hidrógeno, que queda en espera del sexto átomo de hidrógeno.
- (7) *Shell 3* aporta un sexto átomo de hidrógeno.
- (8) Se forma una tercera molécula de agua. *Cat* en *shell 4* realiza una 4^{ta} lectura (aporte de oxígeno).
- (9) El <control-C> genera una señal. Esto hará que *c_wait* (ver archivo *kmutex.h*) retorne un valor distinto de 0 (sin haber adquirido el mutex). Ante esto Ud. debe hacer que *read* retorne 0 bytes leídos, lo que *cat* interpretará como el fin de archivo y por lo tanto terminará.

Recursos

Baje de U-cursos el archivo *modules2019-2.tgz*. Descomprímalo con el comando:

```
$ tar zxvf modules2019-2.tgz
```

Contiene enunciados y soluciones de tareas de semestres anteriores con instrucciones para compilarlas y ejecutarlas (ver archivos README.txt en cada directorio). Además se adjunta un tutorial sobre programación de módulos y drivers en Linux. Le será de especial utilidad el directorio *Syncread* con el enunciado y la solución de la tarea 3 del semestre otoño de 2013.

Programa su solución en el archivo *h2o-impl.c* del directorio *H2o*. Ahí encontrará un *Makefile* para compilar su tarea y las instrucciones para crear el dispositivo */dev/h2o*. Resuelva su tarea usando los mutex y condiciones incluidos en el directorio *H2o*. Como ejemplo de utilización estudie la solución de *Syncread*. Estos mutex y condiciones son análogos a los de pthreads y están implementados a partir de los semáforos del núcleo de Linux en el archivo *kmutex.c* con encabezados en *kmutex.h*.

Antes de cargar y probar su tarea asegúrese de ejecutar el comando Unix *sync* para garantizar que sus archivos hayan sido grabados en disco y no están pendientes en un caché de Unix. Recuerde que los errores en su driver pueden hacer que Linux se bloquee indefinidamente y tenga que reiniciar el sistema operativo.

Entrega

La tarea se entrega *funcionando* en U-cursos. Para ello entregue solo el archivo *h2o-impl.c* que implementa el driver pedido. Se descontará medio punto por día de atraso, excepto días sábado, domingo o festivos.