

# WUOLAH



Info\_sw

[www.wuolah.com/student/Info\\_sw](http://www.wuolah.com/student/Info_sw)



763

## APSO\_apuntes\_practica6.pdf

*APSO. Parte 2: programación*



**2º Administración y Programación de Sistemas Operativos**



**Grado en Ingeniería Informática**



**Escuela Técnica Superior de Ingeniería  
UHU - Universidad de Huelva**

 **escuela  
de negocios**  
CÁMARA DE SEVILLA

## MÁSTER EN DIRECCIÓN Y GESTIÓN DE RECURSOS HUMANOS

[www.mastersevilla.com](http://www.mastersevilla.com)

Inscríbete



**BECAS**

## APSO. PROGRAMACIÓN DEL SISTEMA.

### APUNTES PRÁCTICA 6:

A diferencia que las FIFOs y las TUBERÍAS (que usan memoria del disco), las colas de mensajes se almacenan en memoria RAM.

//NOTA: ¡¡HAY QUE ELIMINAR TODAS LAS COLAS DE MENSAJES!! -> La primera vez que un profesor detecte

que hemos dejado en memoria RAM las colas de mensajes recibiremos un aviso, pero la segunda vez que se

detecte que hay alguna cola no borrada, nos inhabilitan la cuenta de Polifemo.

En una cola de mensajes puedo guardar registros (struct) de diferentes tipos, y puedo leer posterior-

mente un determinado tipo de registro. Las estructuras tendrán un primer campo obligatorio (tipo long):

```
struct Mensaje1
{
    Long tipo;
    ...
    int b;
}
```

\*\*\*\*\*

ftok(ruta\_fichero,número\_entero); -> Hay que hacerlo previamente a la creación de la cola de mensajes.

La función 'ftok' devuelve una clave(key).



NOTA: 'ftok' LO DEBEN HACER TODOS LOS PROCESOS QUE HAGAN USO DE LA COLA DE MENSAJES, CON LOS MISMOS PARÁMETROS.

PARÁMETROS:

ruta\_fichero: vale cualquier fichero. Normalmente se pone una ruta relativa a un fichero de nuestro

directorio. Es muy recomendable poner un fichero nuestro (por ejemplo, nuestro Makefile o incluso

mejor un fichero creado para este único fin), y no uno genérico a todos los usuarios, ya que si

ponemos uno que coincida con el que pone otro usuario, ambos usuarios estaríamos accediendo a la

misma cola de mensajes (podemos tener muchos problemas si ocurre esto).

número\_entero: el número que sea, aleatorio. Por ejemplo, un 18, un .

\*\*\*\*\*

```
msgget(clave_devuelta_por_ftok, 0600 | IPC_CREAT);
```

'NOTA: 'msgget' LO DEBEN HACER TODOS LOS PROCESOS QUE HAGAN USO DE LA COLA DE MENSAJES, CON LOS MISMOS PARÁMETROS.

PARÁMETROS:

Primer parámetro: la clave devuelta por el ftok.

Segundo parámetro: '0600 | IPC\_CREAT' -> permisos + opción 'IPC\_CREAT' que crea la cola si no está

creada y no hace nada si ya ha sido creada previamente.

\*\*\*\*\*

ELIMINACIÓN O DESTRUCCIÓN DE UNA COLA DE MENSAJES:

¡ATENCIÓN! -> ESTO SÓLO DEBE HACERLO UNO DE LOS PROCESOS QUE USEN LA COLA DE MENSAJES.

```
msgctl(lo_que_devuelve_msgget,IPC_RMID,(struct msqid_ds *)NULL);
```

\*\*\*\*\*

ES ACONSEJABLE COMPROBAR SI HEMOS DEJADO COLAS CREADAS AL ROMPER (CTRL+C) DURANTE LA EJECUCIÓN UN PROCESO.

HACERLO SIEMPRE TRAS TRABAJAR CON POLIFEMO PARA ASEGURARNOS QUE NO DEJAMOS COLAS CREADAS EN RAM.

```
ipcs -q //muestra las colas de mensajes que tenemos activas en nuestra sesión.
```

Si tenemos alguna activa, debemos borrarla!!

```
ipcrm msg identificador_de_la_cola //borra una cola
```

También hay disponible un script llamado "borracolas" que creó Pedro Abad que sirve para borrar las

colas del tirón. Buscar el script con el find y copiarlo. Esto en el examen no lo podemos hacer, por lo

que es aconsejable usar lo anterior.

\*\*\*\*\*

ENVÍO DE MENSAJES A UNA COLA:

`msgsnd ( int msqid, struct msgbuf *msgp, int msgsz, int msgflg );` //ver ejemplo en apuntes

`int msqid:` lo que devuelve `ms..get`

RECEPCIÓN DE MENSAJES: si al leer no hay nada escrito en la cola de mensajes, se espera a que haya

algo (igual que las FIFOs y PIPES).

`msgrcv (lo_que_devuelve_msgget, (struct msgbuf *)&objeto_mensaje,`  
`sizeof(objeto_mensaje)-sizeof(long),tipo_mensaje,0)` //ver ejemplo en apuntes

'tipo\_mensaje' es el identificador del struct del que queremos leer (1, 2, 3...)

El último param. se pondrá '0' si queremos que se espere a que haya algo en la cola de mensajes para leer (RECOMENDABLE).

\*\*\*\*\*

GENERAR CARACTERES ALEATORIOS:

TENGO UN ARRAY CON TODOS LOS CARACTERES DEL ABECEDARIO.

GENERO UN NÚMERO ALEATORIO ENTRE 0 Y 27 (Nº DE CARACTERES DEL ABECEDARIO -1)

VOY AL ARRAY Y SACO LA LETRA DE ESA POSICIÓN ALEATORIA DEL ARRAY.