Sistemas Operativos. Práctica 3.

Pablo Cuesta Sierra y Álvaro Zamanillo Sáez

Índice

Memoria compratida

Ejercicio 1

a) Primero (línea 2), se intenta crear un segmento de memoria compartida, con las opción `O_EXCL`, si existe el segmento, `shm_open` devuelva error (-1). Después (línea 3), se trata este error y, en caso de que el error se deba a que el segmento con ese nombre ya existía, se abre este segmento (esta vez sin usar la flag de crear). Si el error no era porque ya existiera o si al intentar abrir el segmento ya existente hay otro error, se llama a `perror` y termina el programa.

b) Para forzar la inicialización (en la primera llamada), habría que quitar la opción `O_EXCL` (para que si existe, no se devuelva error), y añadir la opción `O_TRUNC`, con la que, si existe, se trunca a tamaño 0. De este modo, se forzaría la inicialización.

Ejercicio 2

a) Para obtener el tamaño basta con:

```
if (fstat(fd, &statbuf) != 0) {
    perror("fstat");
    exit(EXIT_FAILURE);
}
```

Y tras esto, el tamaño se encuentra en `statbuf.st_size`.

b) Para truncar a 5 bytes:

```
if (ftruncate(fd, 5) != 0) {
    perror("ftruncate");
    exit(EXIT_FAILURE);
}
```

El fichero resultante contiene: `Test␣`. Es decir, solamente los 5 primeros caracteres.

Ejercicio 3

a) Cada vez que se ejecuta el programa, el counter incrementa su valor en 1. Esto es porque el programa abre el fichero en caso de que ya exista, y varía el valor que ya existe. Si borramos el fichero antes de ejecutar,

el fichero se crea de nuevo y tras la ejecución el valor vuelve a ser 1.

b) El fichero es binario, por lo que las variables no aparecen con los valores en forma de caracteres que se puedan leer con un editor de texto.