Universidad Católica
San Pablo

Ciencia de la Computación

Sistemas Operativos

Trabajo Final de Laboratorio

Alumno: Justo Alfredo Perez Choque

Profesor: Julio Omar Santisteban Pablo
Grupo: CCOMP6-1.1

Semestre VI
2023-1

"Los alumnos declaran haber realizado el presente trabajo de acuerdo a las normas de la Universidad Católica San Pablo"

1 Explicación de Lógica

1.1 Archivo *modulo.c*

Contiene el código hecho en C para crear el módulo. Extraido en su mayor parte del archivo pdf proporcionado por el profesor *Writing a Linux Kernel Module — Part 2 Character Device.pdf*.

El cambio más importate fue una linea en la función *device write*:

```
1 memset(msg, '\0', BUF_LEN);
```

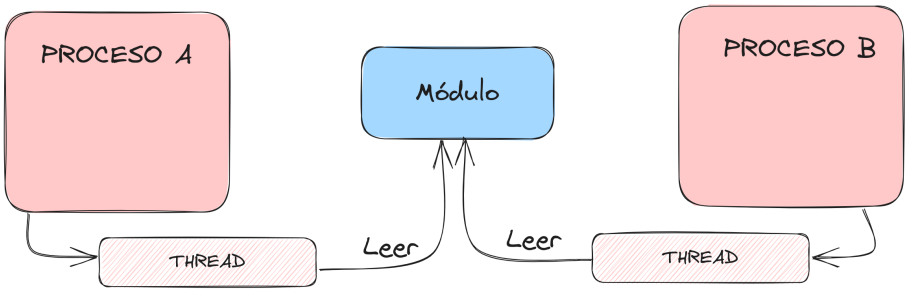
que limpiará array de chars antes de escribir un nuevo mensaje.
Después sigue la misma lógica explicada en dicho PDF y explicada en clase: Crear el módulo.

1.2 Archivo *test.cpp*

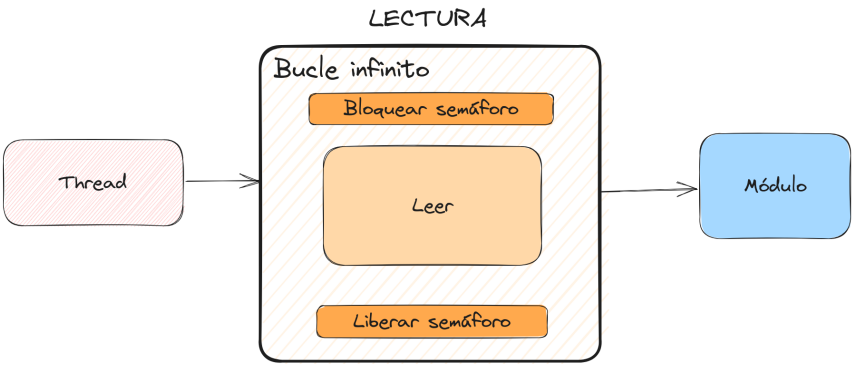
Para la implementación se usaron threads y semáforos. A continuación una rápida explicación del código.

1.2.1 Lectura

Cada proceso tendrá un thread que estará en un bucle infinito leyendo el módulo.

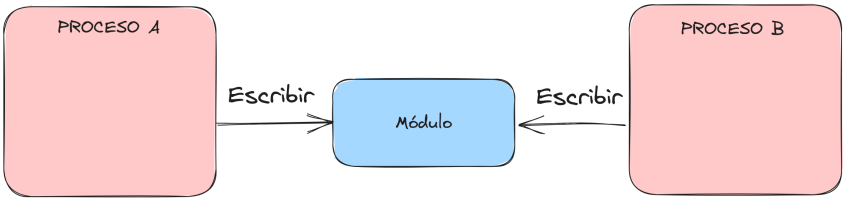


En el bucle infinito del thread, lo primero que hará será bloquear el semáforo para asegurarse de ser el único que esta accediendo al módulo.



1.2.2 Escritura

Para la escritura el mismo programa recibirá la entrada, (no bloqueará el output por si otro proceso llega a escribir) y al igual que la lectura: bloqueará el semáforo, escribirá y liberará el semáforo.



Y con esta lógica permite ejecutar cualquier cantidad de programas y realizar un chat sincronizado.

2 Ejecución

Seguir los siguientes pasos para la creación del módulo:

a) Para tener privilegios: *sudo su*
b) Compilar: *make*.
c) Instalar el módulo: *insmod modulo.ko*
d) Visualizar los mensajes del kernel: *dmesg*.
e) Buscar el mensaje de ucsp, ejecutar el comando que nos coloca. Ejemplo: *mknod /dev/ucsp c 509 0*.

Seguir los siguientes pasos para la ejecución del test:

a) Compilar el test: *g++ -Wall -o test.exe test.cpp*
b) Ejecutar el test: *./test.exe*

El ejecutable test se puede ejecutar las veces que se desee.