

Sistemas Operativos SIU4085 Proyecto 25% Sistema para el Préstamo de Libros Marzo, 2025

# Sistema para el Préstamo de Libros

## 1. Objetivos principales

- a. Resolver un problema utilizando procesos e hilos de la biblioteca **POSIX**.
- b. Emplear mecanismos de sincronización de procesos y comunicación de procesos usando pipes.
- c. Utilizar de forma correcta llamadas al sistema relacionadas con hilos y procesos.

### 2. Descripción

El objetivo del proyecto es desarrollar un sistema de préstamo de libros. Los componentes del sistema se describen a continuación:

**Procesos Solicitantes (PS):** son los procesos que ingresarán al sistema como las operaciones sobre los libros que solicitan los usuarios. Las <u>operaciones</u> son:

- 1. **Devolver un libro:** permite a un usuario devolver un determinado libro.
- 2. **Renovar un libro:** permite a un usuario renovar, por una semana adicional, un libro que ya posee.
- 3. Solicitar prestado un libro: se solicita el préstamo de un determinado libro.

Los **PS** pueden generar requerimientos de estos tres tipos de dos formas:

- A través de un menú.
- ii. Provenientes de un archivo de texto previamente configurado, de manera que se lean las peticiones de forma automática. Se supone que el formato del archivo estará correcto, pero puede ser que un usuario pida prestado un libro que no exista, en este caso su sistema debe manejar adecuadamente esta situación.

**Receptor de Peticiones (RP):** este proceso recibe las peticiones de los PS y, dependiendo del tipo de solicitud, realiza las siguientes <u>acciones</u>:

4. Devolver un libro: en este caso, el receptor RP acepta de forma inmediata la operación y le devuelve al PS correspondiente, una respuesta positiva indicando que la biblioteca está recibiendo el libro. Una vez que le ha respondido al PS, el Receptor <u>publica en un buffer</u> un requerimiento de Devolución, que será atendido por un hilo <u>auxiliar1</u> que hace parte de este proceso. Este hilo <u>auxiliar1</u> tomará el requerimiento y realizará la actualización correspondiente en la Base de Datos (BD), donde se guarda toda la información de los libros. Si bien utilizaremos un archivo de texto para almacenar la información de los libros de la biblioteca, en este documento diremos siempre que es la Base de Datos (BD).

- 5. Renovar un libro: al igual que en el caso anterior, el receptor RP acepta de forma inmediata la operación y le devuelve al PS una respuesta positiva indicando la nueva fecha de entrega (1 semana a partir de la fecha actual). Una vez que le ha respondido al proceso PS, el RP <u>publica en un buffer</u> un requerimiento de <u>Renovación</u>, que será atendido por el mismo hilo <u>auxiliar1</u> mencionado anteriormente. Este hilo <u>auxiliar1</u> tomará el requerimiento y realizará la actualización en la BD (se actualiza la fecha de devolución del libro).
- 6. Solicitar prestado un libro: como en los dos casos anteriores, el PS hará el requerimiento de un libro y esperará a que el RP le dé respuesta. En este caso el receptor NO utiliza el hilo auxiliar1 sino que verifica, él mismo, en la BD la disponibilidad del libro solicitado, y si está disponible le envía una respuesta positiva al PS realizando las actualizaciones correspondientes en la BD. Si no está disponible, envía al PS un mensaje con una respuesta negativa.
- 7. **Aviso de terminación del proceso solicitante:** El proceso solicitante antes de terminar puede enviar un <u>mensaje</u> especial al **RP**, indicándole que ya no va a enviar más solicitudes. El <u>mensaje</u> puede estar en el archivo de solicitudes o ser escrito a partir del menú interactivo del proceso solicitante **PS**.
- 8. El **RP** al comenzar a ejecutarse generará un hilo *auxiliar2*, que podrá recibir comandos de la consola desde la que se invocó el **RP**. En el proyecto, se ha planificado el uso de 2 comandos diferentes, a saber:
  - Comando "s" por salida que termina la ejecución de los hilos auxiliares y del RP.
  - Comando "x" por reporte que genera un reporte en el que se muestran por pantalla los datos de los ejemplares de los libros que han sido prestados, los que han sido renovados y los que han sido devueltos. El formato debe presentarse de la siguiente manera:

#### Status, Nombre del Libro, ISBN, ejemplar, fecha

9. Ejemplo de un reporte:

```
P, Cálculo Diferencial, 1200, 1, 1-10-2021
```

P, Cálculo Diferencial, 1200, 2, 1-10-2021

P, Redes de Computadores, 3400, 1, 1-10-2021

P, Operating Systems, 2233, 1, 1-10-2021

P, Operating Systems, 2233, 1, 1-10-2021

D, Operating Systems, 2233, 1, 1-10-2021

D, Cálculo Diferencial, 1200, 2, 1-10-2021

Los datos reportados deben reflejar el estado de las operaciones que hacen los procesos que acceden a la **BD**. En el caso de la opción <u>verbose</u> todas las operaciones recibidas por el proceso **RP** deben poder verse por la consola del computador donde se invocó ese proceso.

### 3. Detalles de Implementación

### Procesos Solicitantes (PS)

Los procesos solicitantes se invocarán desde uno o varios terminales de la siguiente forma:

```
$solicitante [-i file] -p pipeReceptor
```

#### Donde:

- Lo que va entre corchetes es opcional.
- El flag -i solo se indica si la entrada del proceso vendrá de un archivo. A continuación del flag se escribe el nombre del archivo.
- El flag -p indica que a continuación viene el nombre del pipe nominal que permitirá la comunicación entre procesos.
- pipeReceptor es el nombre del pipe que servirá de comunicación entre el RP y el PS. Todos los PS enviarán sus solicitudes por este pipe (ver Figura 2).

#### Entrada a través de Archivo

En caso de que la entrada del **PS** se lea desde un archivo, este será un archivo *texto* con el siguiente formato:

```
Operación, nombre del libro, ISBN.
```

Donde Operación son los caracteres D, R y P para los tres tipos de operaciones. El ISBN es un número entero que identifica de forma unívoca a los libros. Los tres campos van separados por comas. Ejemplo de un archivo de entrada:

- D, Alicia en el País de las Maravillas, 2133
- R, Hamlet, 234
- R, Historia de Colombia, 223
- P, Cálculo Diferencial, 120
- P, Algebra Lineal, 111
- Q, Salir, 0

**Nota:** Debe respetar el formato de este archivo, ya que los archivos de prueba serán provistos por el(la) profesor(a) el día de la sustentación.

Al terminar de leer el archivo y enviar las solicitudes el proceso **PS** queda a la espera de las respuestas del **RP** y podrá terminar su ejecución si recibe el comando se para salir.

#### Entrada a través del menú

En este caso, el proceso ofrecerá un menú al usuario donde le pregunte: <u>operación a realizar</u>, <u>nombre del libro</u> e <u>ISBN</u>. Ante la operación solicitada y según lo especificado en la sección anterior, el usuario recibirá una respuesta. El menú le dará al usuario la opción de continuar introduciendo operaciones o de salir. Si el usuario decide salir se termina el programa.

### Proceso Receptor (RP)

El Proceso Receptor se invocará desde un terminal de la siguiente forma:

\$./receptor -p pipeReceptor -f filedatos [-v ] [ -s filesalida ]

#### Donde:

- -p flag que indica que a continuación viene el nombre del pipe nominal que permitirá la comunicación entre los procesos.
- pipeReceptor es el nombre del pipe que servirá de comunicación entre los **PS** y el **RP**. Todos los **PS** enviarán sus solicitudes por este pipe (ver Figura 2).
- -f flag que indica que el siguiente parámetro es el nombre de un archivo.
- filedatos es el nombre del archivo donde se encuentra la **BD** inicial que maneja el sistema (la cantidad de libros y sus características). Se les proveerá archivos de ejemplo y el archivo final para el día de la sustentación.
- -v Esta opción <u>verbose</u>, activa la característica de que todas las solicitudes sobre libros de préstamos, renovaciones o devoluciones se puedan observar en la consola del computador desde la cual se invocó al **RP**.
- -s flag que indica que a continuación viene el nombre del archivo de salida.
- filesalida cuando el receptor detecte que todos los PS han terminado, el RP termina y escribe en este archivo como quedó la BD: cuál es el estado de cada libro, cuantos ejemplares están disponibles de cada libro, y si están prestados presenta la fecha de devolución.

Como se indicó en la sección **Descripción**, el **RP** tiene un hilo interno que se encarga de las operaciones de devolución y renovación. El **RP** colocará estas solicitudes en un buffer de tamaño N, de donde serán tomadas por el hilo *auxiliar*. Esta funcionalidad se implementará como Productor/Consumidor. El valor de N puede ser una constante y en la sustentación se probará con N <= 10. La Figura 1, presenta la comunicación entre el RP y hilo *auxiliar*.



Figura 1. Comunicación entre el RP y el hilo auxiliar.

#### Comunicación entre Procesos

La comunicación entre los **PS** y el **PR** será por medio de pipes (ver Figura 2).

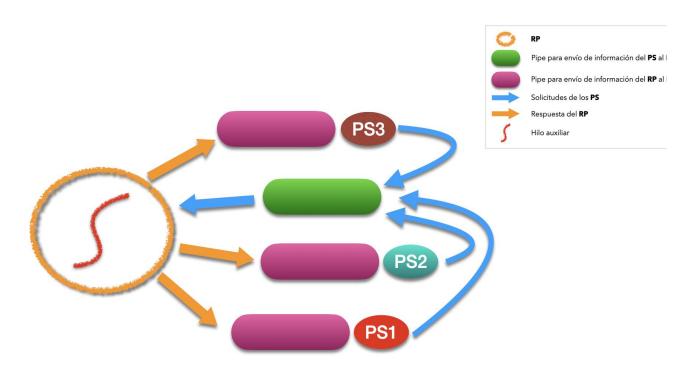


Figura 2. Comunicación entre los procesos solicitantes PS y el receptor de procesos RP.

#### Base de Datos

La base de datos puede ser un archivo de texto o una base de datos local simple (una sola tabla). Si es un archivo de texto, el archivo tiene la información de cada libro almacenada en la siguiente forma:

```
nombre del libro, ISBN, numero ejemplares ejemplar1, status, fecha ejemplar2, status, fecha ...
```

En la primera línea se encuentra el nombre del libro, el ISBN y el número de ejemplares que tiene la biblioteca. En las líneas siguientes se encuentra la información de cada ejemplar: el status indica si está prestado (P) o disponible (D). En todos los casos, la fecha será la fecha actual.

Ejemplo:

```
Operating Systems, 2233, 4

1, D, 1-10-2021

2, D, 1-10-2021

3, P, 1-10-2021

4, P, 1-10-2021

Data Bases, 2234, 2

1, P, 1-10-2021

2, P, 1-10-2021

Programming Languages, 2240, 1

1, D, 1-10-2021
```

Ud, puede agregar los campos que considere convenientes para representar cada libro. Por simplicidad para la implementación del proyecto, se permitirá que los ISBN sean más cortos que los reales y no se tomará en cuenta los guiones o las reglas de formación del ISBN, pero si se debe tomar en cuenta el tamaño máximo de caracteres del formato de ISBN.

# 4. Entrega y Sustentación

El día de la sustentación, los profesores entregan archivos de entrada y casos de prueba a ser usados. La entrega que deben realizar en la plataforma del aula virtual consiste en:

- Códigos fuente de las funcionalidades implementadas.
- Archivo makefile para generar los procesos ejecutables.
- Archivo pdf como informe de proyecto. Este archivo se usará para explicar la estructura del sistema.

Cualquier sospecha de copia o fraude académico será elevada a las instancias correspondientes dentro de la Facultad de Ingeniería.

## 5. Rubricas de Evaluación

CATEGORÍA RÚBRICA	Excelente	Regular	Deficiente
<u>A.</u>	4	[2,4)	[0 , 2)
<u>B.</u>	9	[5 , 9)	[0 , 5)
<u>C.</u>	5	[2.5 , 5)	[0 , 2.5)
<u>D.</u>	7	[3.5 , 7)	[0 , 3.5)
Total	25		

CATEGORÍA	Excelente	Regular	Deficiente		
A. COMPRENDER EL FUNCIONAMIENTO DEL PROBLEMA PLANTEADO Y LA IMPLEMENTACIÓN CUMPLE LAS ESPECIFICACIONES PLANTEADAS.	Se realiza la solución del problema según el enunciado, respetando requerimientos de entrada, salida y TODAS las especificaciones para la implementación.	El estudiante realiza la solución al problema siguiendo la mayoría de las especificaciones indicadas en el enunciado, pero tiene algunas fallas menores.	El estudiante no sigue la mayoría de las especificaciones que se dieron en el enunciado con respecto a los argumentos de cada tipo de proceso, las operaciones de los talkers y el manager, comunicación síncrona o asíncrona entre procesos, funcionamiento del proyecto		
B. FUNCIONAMIENTO DEL PROGRAMA	Según el funcionamiento de los casos de prueba, correcta implementación y comunicación entre los procesos.				
<u>C.</u> CÓDIGO	Se siguen todas las indicaciones del enunciado: Uso de Guía de Programación C y archivos de encabezado. Se validan llamadas al sistema. El código está bien documentado.	El código presenta deficiencias menores en no más de dos de los siguientes aspectos: documentación, estructura o validación de las llamadas al sistema.	Se presenta alguna falla importante en 2 o más de los siguientes aspectos:  Documentación  Validación de las llamadas al sistema  Estilo de programación (Estructura)		
<u>D.</u> SUSTENTACIÓN	El estudiante responde correctamente a la(s) pregunta(s) formuladas. En el caso de que se solicite modificación, esta se realiza correctamente en el tiempo establecido. El estudiante demuestra buen manejo de la interfaz de línea de comando y de las herramientas necesarias para ejecutar su proyecto.	El estudiante realiza la modificación solicitada correctamente pero toma más del tiempo establecido o la modificación tiene problemas menores. En el caso de preguntas, el estudiante responde bien a la mayoría de las preguntas formuladas. En general, se observa que el estudiante maneja la interfaz de línea de comandos del SOP, no obstante el equipo tuvo que ayudarlo en algunos momentos de la sustentación.	El estudiante no responde la o (las) preguntas formuladas o no logra el funcionamiento de la modificación solicitada, sólo muestra algunas líneas de código. El manejo de la interfaz de línea de comandos del SOP es deficiente.		