

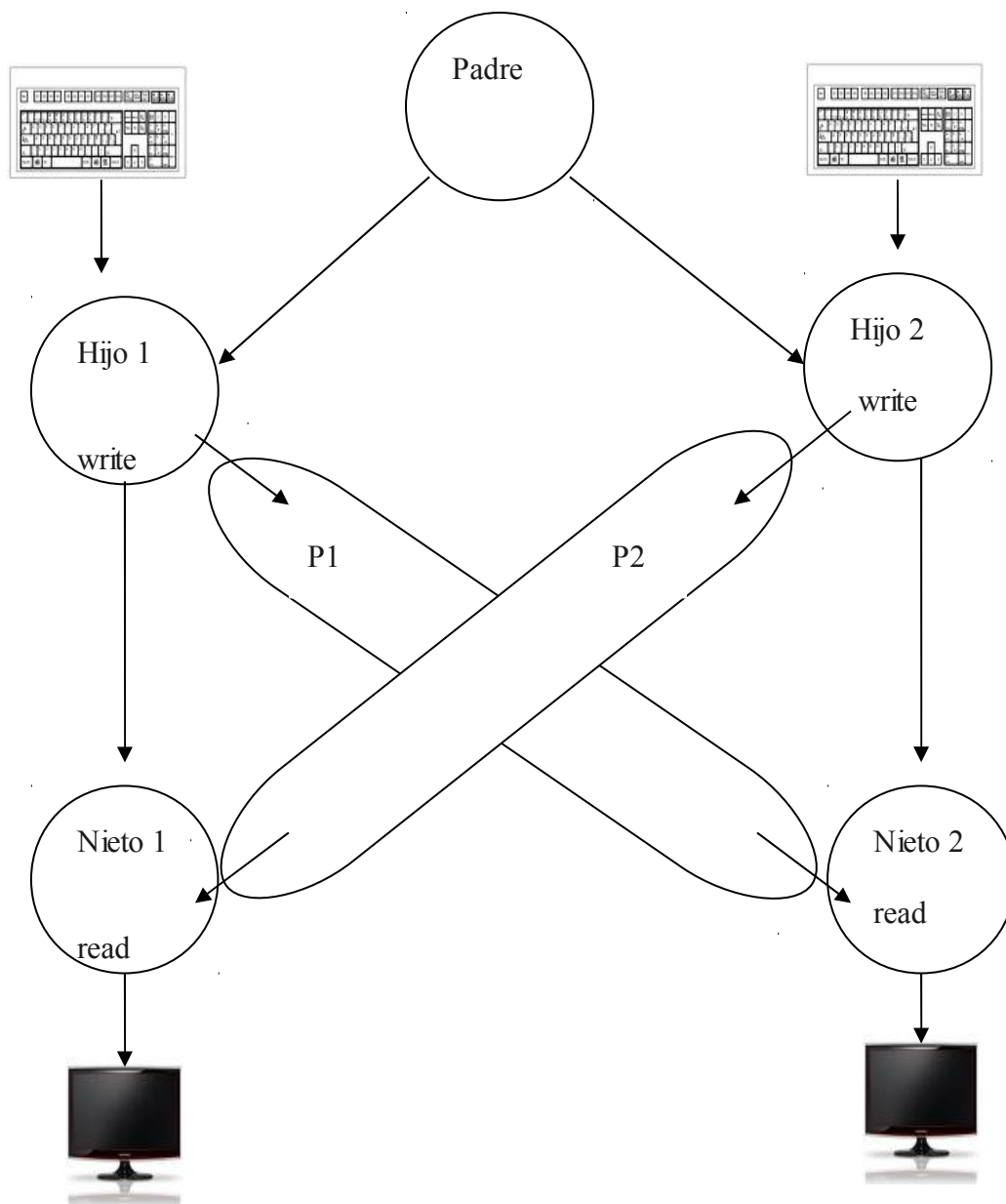
Ejercicio

- Realizar un CHAT entre procesos emparentados usando Pipes.

Objetivos

- Aprender características básicas de la programación concurrente.
- Aprender los mecanismos que usa el Sistema Operativo para la creación de procesos.
- Aprender el uso del recurso Pipe para la comunicación entre procesos emparentados.

Diagrama del Planteo



Descripción del Planteo

Padre: Crea los dos recursos pipe, el pipe P1 y el pipe P2, comienza la creación de la estructura de procesos, especifica que no será escritor ni lector de ninguno de los dos pipes y espera la terminación del hijo 1 y del hijo 2.

Hijo 1: Será únicamente escritor del pipe P1, el mensaje que escribe en el pipe P1, lo obtiene de la entrada estándar. La lectura de la entrada estándar termina cuando obtiene el string “chau” y se queda esperando la terminación del Nieto 1.

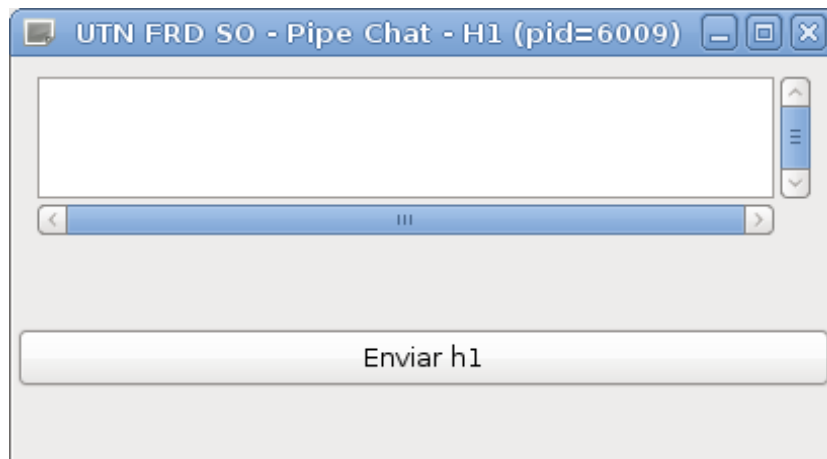
Hijo 2: Será únicamente escritor del pipe P2, el mensaje que escribe en el pipe P2, lo obtiene de la entrada estándar. La lectura de la entrada estándar termina cuando obtiene el string “chau” y se queda esperando la terminación del Nieto 2.

Nieto 1: Será únicamente lector del pipe P2, la lectura del pipe la escribe en la salida estándar, y termina cuando lee del pipe “chau”.

Nieto 2: Será únicamente lector del pipe P1, la lectura del pipe la escribe en la salida estándar, y termina cuando lee del pipe “chau”.

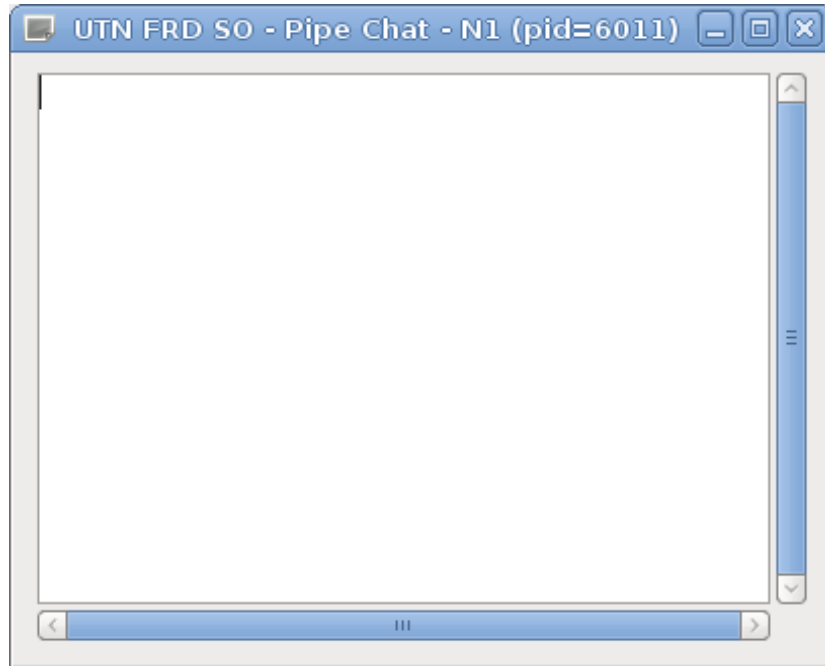
Implementación

Se pide la implementación del ejercicio en un entorno operativo Linux, utilizando el compilador del lenguaje C y la librería gráfica GTK+. De esta forma, se pueden visualizar los procesos distintas ventanas. Por ejemplo el proceso hijo (H1) podría ser:



en el evento open de la ventana (on_window_open_h1) se abre el pipe1; en el evento click del botón (on_bsendh1_clicked) se graba en el pipe1 (si ingresó “chau” se termina la aplicación haciendo un wait() por nieto1 (N1)); en el evento close de la ventana (on_window_destroy_h1) se graba “chau” en el pipe1 y se termina la aplicación haciendo wait() por nieto1 (N1).

y el proceso nieto1 (N1) podría ser:



en donde se recepcionarán todos los mensajes provenientes de hijo2 (H2). En este caso, no es posible implementar toda la lógica del manejo de pipes (puesto que el proceso puede quedar bloqueado y afectaría al aspecto visual de la aplicación) asociado con eventos de ventana de H2, por lo tanto, se optó por crear un thread (hago_n1) el cual recibe una referencia al area de texto de la ventana para su actualización a medida que lee caracteres de pipe2

Requerimientos

Para utilizar GTK+ se requiere (en Debian / Ubuntu) los siguientes paquetes instalados: libc6-dev, pkg-config, libglib2.0-dev, libgtk2.0-dev. El paquete pkg-config nos ayudará en el armado de la compilación y linkedición del programa:

Compilación:

```
$ gcc -Wall -c gtkchatpipe.c `pkg-config --cflags gtk+-2.0 --libs` -lpthread
```

Linkedición:

```
$ gcc -Wall -o gtkchatpipe gtkchatpipe.c `pkg-config --cflags gtk+-2.0 --libs` -lpthread
```

Lic. Juan Carlos Romero
Mg. Guillermo Cherencio