

Proyecto: Monitoreo de Sensores

Objetivo del Proyecto

Después de realizar este proyecto, los estudiantes estarán en capacidad de utilizar herramientas para la comunicación y sincronización de procesos e hilos. Se utilizarán pipes nominales para la comunicación entre procesos y semáforos para sincronizar los hilos. Adicionalmente, emplearemos llamadas al sistema para la manipulación de archivos de texto.

Contexto

El agua es uno de los recursos vitales en nuestro planeta. Con una población en aumento, es importante monitorear la calidad de este recurso tan importante para detectar cambios y tomar las acciones requeridas. La calidad de agua es una medida de qué tan adecuada es una fuente hídrica para un uso específico, por ejemplo, agricultura o generación de energía. Hay varios parámetros esenciales que nos ayudan a medir la calidad del agua, entre ellos el PH, la temperatura, la conductividad, el oxígeno disuelto y la turbidez (<https://www.hannacolombia.com/blog/post/190/guia-para-la-medicion-en-campo-calidad-del-agua>)

En este proyecto realizaremos un sistema sencillo donde se simula la medición de dos parámetros de una reserva de agua a través de sensores: PH y temperatura. Estos parámetros se enviarán a un proceso monitor, que los almacenará adecuadamente y avisará al usuario si se genera alguna alerta con los indicadores medidos.

Descripción General del Sistema a Desarrollar

A continuación, se explica cada uno de los componentes de la arquitectura de software (figura 1):

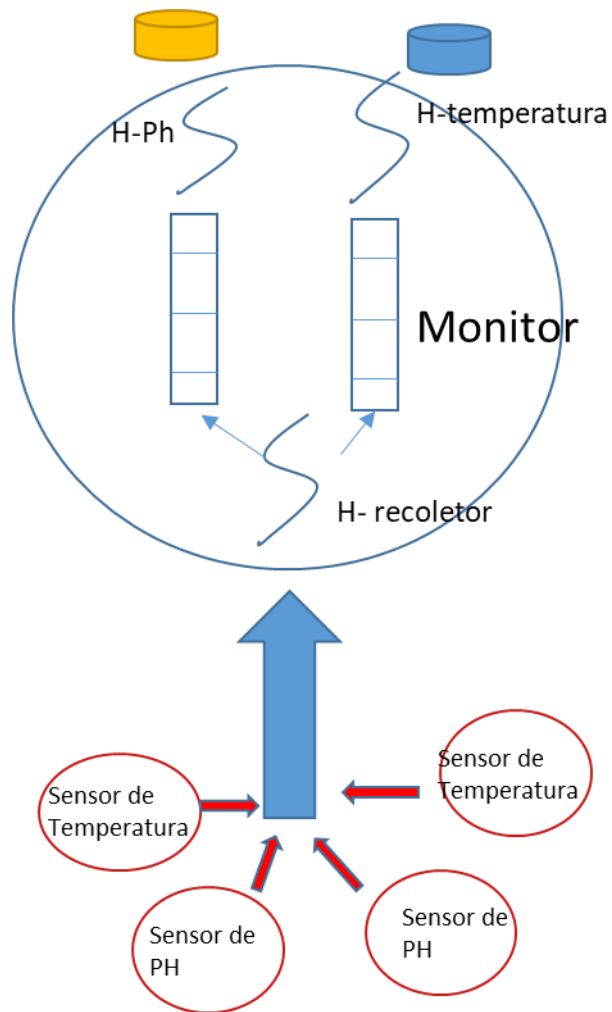


Figura 1: Procesos/hilos y mecanismos de comunicación

Sensores

Los sensores que miden la temperatura y el PH serán simulados por procesos. Los procesos sensores se invocarán desde el shell de la siguiente forma:

\$./sensor -s tipo-sensor -t tiempo -f archivo -p pipe nominal

Donde:

tipo_sensor: Dado que cada proceso sensor solo reportará los valores de una variable, se utilizará este parámetro para especificar si el proceso reportará PH o temperatura. Se utilizará el número 1 para temperatura y el número 2 para el PH.

tiempo: este parámetro indica cada cuanto tiempo (t) se va a enviar la medición del proceso sensor al proceso monitor. Los valores de las variables se van leyendo desde un archivo.

Cada proceso sensor, lee una medida del archivo, **espera t segundos** y luego envía la siguiente medición al monitor.

archivo: es el nombre del archivo con medidas de temperatura o de PH, según sea el caso. El archivo tendrá tres tipos de valores. a) Valores correctos y dentro de los rangos establecidos en la tabla 1. b) Valores positivos, pero por fuera de los rangos establecidos en la tabla (estos valores generarán alertas) c) Valores negativos: estos valores se consideran erróneos, por tanto, serán descartados al ser recibidos por el monitor.

Ejemplo de posibles valores en el archivo con medidas de temperatura:

68
69
71
70
-2
90
50

Ejemplo de posibles valores en archivo con medidas de PH:

6.0
6.5
-3
4.0
7.0
7.2

pipe-nominal: es el nombre del pipe nominal que permite la comunicación entre los sensores y el monitor.

Nota 1: Pueden usar el mismo archivo para todos los sensores del mismo tipo o utilizar diferentes archivos. Estos archivos pueden ser de cualquier tamaño. Cada medición va en una línea distinta.

Nota 2: pueden y deben ejecutarse varios sensores al tiempo. Aunque en la figura 1 se observan dos sensores de cada tipo, se pueden crear más de dos procesos sensores en una corrida del proyecto. Todos los procesos se ejecutarán de forma concurrente.

Nota 3: la implementación de los flags **-s**, **-t**, etc. antes de los parámetros de monitores y sensores es obligatoria. La función de los flags es poder cambiar el orden de los parámetros. Por ejemplo, las siguientes dos invocaciones son correctas:

```
$ ./sensor -s 1 -t 3 -f datos -p pipe1
```

```
$ ./sensor -t 10 -p mypipe -s 2 -f file-input.txt
```

Tabla 1: Rangos de los parámetros de calidad.

Parámetro	Valor mínimo	Valor máximo
Temperatura	20°C	31.6°C
PH	6.0	8.0

Monitor

Es el proceso que recibe las medidas de los sensores. El monitor estará internamente conformado por tres hilos (ver figura 1). La función de cada uno de los hilos se describe a continuación:

H-recolector: recibe los dos tipos de mediciones del pipe nominal y las coloca en uno de los dos búferes dependiendo del tipo de medida (PH o temperatura). Si la medida recibida por el pipe es errónea (un valor negativo), se descarta y no se coloca en ninguno de los dos búferes; en este caso el recolector debe imprimir un mensaje por la consola. Cuando el recolector detecte que no hay ningún sensor conectado, esperará 10 segundos antes de enviar un mensaje al resto de los hilos indicando que ya terminó el envío de indicadores. Una vez colocado este mensaje en cada buffer, el hilo recolector elimina el pipe nominal, imprime un mensaje de finalización del procesamiento de medidas y termina.

H-ph: Este hilo recoge las medidas que son colocadas en su buffer (medidas de PH). Todas las medidas recibidas se escribirán en el archivo *file-ph*, cuyo nombre recibe el monitor como argumento de entrada. El hilo H-ph, antes de escribir una determinada medida en el archivo, le anexará la hora actual. Si la medida está fuera de los rangos, el hilo imprime un mensaje de alerta por la consola. Cuando este hilo recibe un mensaje de finalización por parte del hilo recolector, cierra el archivo y termina.

H-temperatura: Este hilo recoge las medidas que son colocadas en su buffer (medidas de temperatura). Todas las medidas recibidas se escriben en el archivo *file-temp*, cuyo nombre recibe el monitor como argumento de entrada. A cada medida se le anexa la hora actual antes de escribirla. Si la medida está fuera de los rangos, el hilo imprime un mensaje de alerta por la consola. Cuando este hilo recibe un mensaje de finalización por parte del hilo recolector, cierra el archivo y termina.

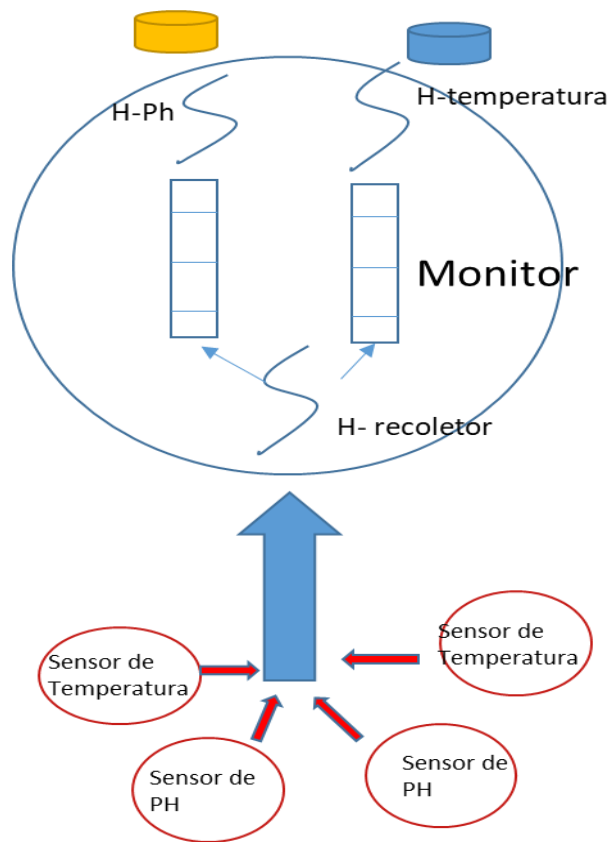


Figura 1: Procesos/hilos y mecanismos de comunicación

El proceso monitor se invocará desde el shell con los siguientes parámetros:

./monitor -b tam_buffer -t file-temp -h file-ph -p pipe-nominal

Donde:

tam-buffer: es el tamaño de los búferes donde el H-recolector colocará las medidas.

file_temp: es el nombre del archivo de texto donde el hilo H-temperatura colocará las mediciones de temperatura recibidas.

file_ph: es el nombre del archivo de texto donde el hilo H-Ph colocará las mediciones de PH recibidas.

pipe-nominal: es el nombre del pipe nominal que permite la comunicación entre los sensores y el monitor.

Mecanismos de sincronización y comunicación

Comunicación entre hilos

El hilo recolector enviará información a los otros dos hilos usando el **patrón productor/consumidor con buffer acotado**. Cada tipo de medida se colocará en uno de los dos búferes. El tamaño del buffer es un argumento de entrada del proceso monitor. Se implementará el patrón según lo visto en clase, es decir, los hilos consumidores se bloquean si el buffer está vacío y el hilo productor se debe bloquear cuando no hay más espacio en el buffer. Los procesos se sincronizarán utilizando los semáforos de la librería POSIX y las operaciones sobre los semáforos vistas en la clase.

Comunicación entre procesos

Los sensores envían información al monitor a través de un **pipe nominal**. El nombre del pipe nominal lo reciben ambos tipos de procesos al momento de su creación. Los sensores solo envían medidas por el pipe, no reciben respuestas.

Consideraciones de Diseño e Implementación

- El proyecto lo deben realizar en lenguaje C o C++ utilizando las llamadas al sistema y llamadas a la librería de hilos y semáforos vistas en clase.
- Deben respetar **todas** las especificaciones de este enunciado.

Sobre la Entrega

- El proyecto lo deben realizar en grupos de dos o tres personas. **No se recibirán entregas individuales.**
- **Entre las semanas 11 a 13 cada equipo debe enseñar a su profesor avances de su proyecto. Para esa fecha, los estudiantes pudieran tener implementada una versión preliminar del proceso Monitor con los tres hilos y los dos búferes. Cada profesor definirá lo que los equipos deben mostrar y el valor de esa primera entrega.**
- Lo deben entregar y sustentar **el domingo 19 de mayo de 2024**. La entrega, que se realizará por BS **antes de las 12 de la medianoche**. Antes de hacer la sustentación se revisará que los estudiantes hayan subido el código en BS.
- La entrega consiste en los códigos fuente y el archivo makefile para hacer la compilación de todo el proyecto. Todos estos archivos deben entregarse en un solo archivo con formato tgz (primero empaquetarlos con tar y luego comprimir con el comando gzip).
- El código debe seguir los lineamientos de la Guía de Programación de C (Esta se encuentra en la Unidad Proyecto del curso/Herramientas para el proyecto). Debe validar llamadas al sistema y que las entradas del usuario se encuentren entre los valores o rangos correspondientes.

- El día de la sustentación se pedirá que cada integrante del grupo demuestre conocimiento sobre el proyecto. Esto se puede hacer solicitando una modificación (individual) o a través de preguntas que deben contestar de forma individual. Todos los integrantes deben estar presentes. Deben seguir todas las especificaciones que este enunciado. **El proyecto se debe compilar y ejecutar desde la interfaz de línea de comando (CLI).**
- Si tiene alguna duda sobre este enunciado diríjase a los Profesores con la debida anticipación.

Mariela Curiel/John Jairo Corredor/Osberth Luef de Castro.

Suerte!