



## Sistemas Operativos

### Taller T21 – Mutex – Semáforos – Pipes - Señales

**Objetivo:** comprender e iniciar el uso más básico de procesos e hilos.

---

Instrucciones:

1. **Grupos:** el taller se realiza en grupos definidos en clase o en la plataforma virtual del curso.
  2. **Entrega:** la entrega consiste en un informe en PDF que se sube al buzón asignado en la plataforma virtual del curso. Este informe debe contener todas las evidencias: respuestas a preguntas, resultados de los experimentos o ejecuciones solicitadas en el orden solicitado, resultados de pruebas y análisis solicitados, incluidas capturas de pantalla asociadas. Si el taller requiere el desarrollo de código, el informe y todos los archivos de código, texto o de resultado de ejecución deben incluirse en un archivo comprimido (zip ó 7z).
  3. **Plagio:** Cualquier evidencia de plagio o copia tiene como consecuencia calificación de cero puntos en la asignación. Toda entrega será analizada con herramientas anti-plagio.
  4. **Plataforma:** todas las operaciones de este taller se realizarán bajo plataforma Linux, en cualquier forma (Linux instalado en computador con doble boot ó Linux instalado en máquina virtual). Si Ud. no dispone de alguna de las opciones, puede usar como último recurso la herramienta <https://cocalc.com/> y crear una cuenta, crear un proyecto, cargar archivos, y usar un terminal de Linux.
- 

#### Parte 1: Exclusión Mutua / semáforos y pipes

1. Construya una aplicación multiproceso/multihilo que implemente el siguiente sistema: 4 procesos generan datos climáticos (que provienen de sensores): un proceso genera temperatura (°C), uno genera humedad relativa (%), uno velocidad de viento (m/s), y uno precipitación (mm). El sistema debe mantener actualizado un solo archivo de texto que lleve la situación del clima actualizada de alguna forma. Todos los procesos deben escribir este mismo archivo de texto. Debido a la naturaleza de cada variable, todos los procesos generan las variables a distintos intervalos de tiempo desconocidos (por ejemplo, se toma la velocidad del viento cada  $5 \pm 2$  segundos, pero la humedad relativa cada  $12 \pm 2$  segundos). Estos tiempos deben considerarse indeterminados, pues un sensor no conoce la existencia de otro. Programe el sistema de 2 formas:
  - a. Como un solo programa multihilo (cada sensor/proceso es un hilo) con sección crítica usando mutex ó semáforos.

- b. Como múltiples programas que se ejecutan por separado que usan pipes nominales/con nombre (FIFO) para entregar sus datos a un proceso central que mantiene el archivo de texto actualizado.
- 

#### **Parte 1: Comunicación entre procesos usando señales.**

Construya una aplicación multiproceso/multihilo que implemente el siguiente sistema: 4 procesos generan datos climáticos (que provienen de sensores): un proceso genera temperatura (°C), uno genera humedad relativa (%), uno velocidad de viento (m/s), y uno precipitación (mm). El sistema debe mantener actualizado un solo archivo de texto que lleve la situación del clima actualizada de alguna forma. Todos los procesos deben escribir este mismo archivo de texto. Debido a la naturaleza de cada variable, todos los procesos generan las variables a distintos intervalos de tiempo desconocidos (por ejemplo, se toma la velocidad del viento cada  $5 \pm 2$  segundos, pero la humedad relativa cada  $12 \pm 2$  segundos). Estos tiempos deben considerarse indeterminados, pues un sensor no conoce la existencia de otro.

Programe el sistema utilizando la siguiente arquitectura: Como múltiples programas que se ejecutan por separado (cada sensor es un programa) que usan pipes nominales/con nombre (FIFO) para entregar sus datos a un proceso central que mantiene el archivo de texto actualizado. Los sensores sólo escriben los datos en el FIFO cuando el proceso central lo solicita enviándoles una señal de tipo **SIGUSR1**. Por ejemplo, cada 5 segundos, el proceso central necesita los datos, y enviará la señal a todos los sensores, para que envíen los datos usando el FIFO.