

# PRÁCTICO 1

## Ejercicio 1 (Obligatorio)

### Objetivo: Estudiar el comportamiento del planificador de Linux

Al final del archivo `/proc/sched_debug` se muestran datos del scheduler para cada proceso listo en el sistema.

Se pide:

1. Explicar el significado de las primeras 6 columnas de esta tabla (incluyendo la que no tiene título) y de la última (tampoco tiene título)
2. Agregar a esta tabla una nueva columna indicando si el nodo del árbol que representa a cada proceso es rojo o negro
3. Realizar pruebas con el programa `loop.c` disponible en el eva. Casos a considerar de la ejecución del programa `loop.c`:
  - Verificar el tiempo de ejecución al ejecutar con prioridades estáticas -20 y 19.
  - Verificar la ejecución de dos procesos del programa (uno con prioridad estática -5 y otro con prioridad estática 5) que corran en forma conjunta. Probar ejecutar ambos programas desde la misma terminal y desde terminales diferentes.

Entregar un informe mostrando los resultados obtenidos y realizando un análisis de los casos pedidos. Examinar para ello los tiempos de ejecución y los datos que se muestran en `sched_debug` durante la ejecución del programa

## Ejercicio 2

### Objetivo: Utilizar y comprender el sistema de colas de trabajo *Work Queue*

La rutina de atención de interrupciones del teclado agenda trabajo a ser realizado en el *bottom half* a través de la *work queue* por defecto del sistema (procesos *kworker* del núcleo).

Esto se puede visualizar en el código: `drivers/input/keyboard/atkbd.c`

Se pide:

1. Modificar el código `atkbd.c` de forma de tener una *work queue* particular para el teclado.

Notas:

- El *bottom half* del teclado se invoca para ciertos eventos especiales. Uno de ellos es al presionar la tecla *Caps Lock*.
- Para la prueba pueden generar unas iteraciones ficticias (`for (..) ;`) dentro de la rutina que se difiere. De esa forma, podrán visualizar a través del comando `top` que proceso de la *work queue* está consumiendo ciclos de procesador.

- Otra posibilidad es mirar la sección de *debugging* de <https://www.kernel.org/doc/Documentation/workqueue.txt> donde se explica como ver que work queue ejecuta el código de cada driver.