

Lab 02

Semáforos en XV6 (RISC-V)

Programando nuestras
propias llamadas al sistema



Hasta Ahora

- **Lab 0:** Interactuamos con programas de usuario (ls, cat, grep, etc) que utilizan syscalls.
- **Lab1:** Implementamos un programa de usuario (mybash) que llama a diferentes syscalls del SO (fork(), wait(), dup(), pipe(), etc).

En este Lab2

- **Lab2:** Implementaremos nuestras propias llamadas al sistema.

Conclusión: con cada lab que hacemos “nos metemos más adentro” en el SO!!!

LAB 2: Semáforos en XV6

Más concretamente

- Vamos a programar un conjunto de syscalls que en particular implementan un sistema de semáforos.
- Y un programa de usuario (muy simple) que utilice dicho sistema de semáforos (ping-pong).
- Todo esto para XV6, un sistema operativo muy pequeño diseñado para fines académicos.
- Poco para escribir, bastante para leer !!!



Qué es un semáforo?

- Es un contador compartido que se inicializa en un valor arbitrario N .
- Un proceso puede incrementarlo (+1) o decrementarlo (-1).
- Si un proceso intenta decrementar el contador cuando su valor es 0, entonces el proceso se bloquea. Si un proceso intenta incrementar el contador cuando su valor es 0, entonces se desbloquean los procesos bloqueados.
- Es decir, un semáforo es un contador con capacidad bloqueante y desbloqueante.
- Se utilizan para coordinar las tareas que llevan a cabo los distintos procesos del sistema.

Syscalls de un Semáforo (espacio de kernel)

sem_open (int sem, int value)

Inicializa un semáforo a un valor arbitrario.

sem_up (int sem)

Incrementa el valor de un semáforo.

sem_down (int sem)

Decrementa el valor de un semáforo. Nunca menor a 0

sem_close (int sem)

Cierra un semáforo del sistema.

Comando pingpong [espacio de usuario]

pingpong int

Programa de usuario que lanza un rally de “ping pong” del tamaño especificado por el entero.

Un proceso genera los “ping” y otro los “pong” => usaremos semáforos para coordinarlos!!!!

Traza válida (rally = 3):

```
$pingpong 3
ping
    pong
ping
    pong
ping
    pong
$
```

Traza no válida (rally = 3):

```
$pingpong 3
ping
    ping
pong
    ping
pong
    pong
$
```

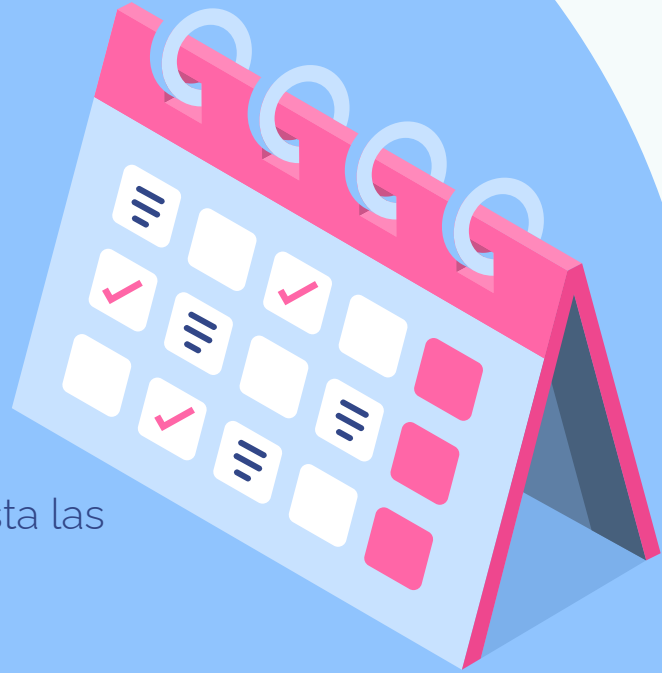
Traza no válida(rally = 3):

```
$pingpong 3
ping
    pong

deadlock!!!
```

Entrega del Lab2

Jueves 28 de Septiembre 2023 hasta las
16:00hs.



Primeros pasos

- Leer el enunciado (hay muchos detalles y ayudas)
- Bajar XV6 aplicar kickstarter patch y hacer el primer commit al repo
- Instalar qemu para RISC-V y correr XV6
- Leer el manual de XV6
- Crear un programa "hello world"
- Crear una syscall "hello world"
- Leer sobre semaforos en OSTEP

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**

Please keep this slide for attribution.

