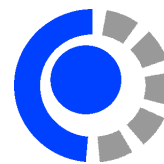




Sistemas Operativos I
Trabajo Práctico Obligatorio 2
2025



Objetivos

- Analizar diferentes algoritmos de **planificación de procesos**.
- Examinar los **estados de un proceso** en un sistema operativo.

Referencias

- [1] Tanenbaum, Bos – Modern Operating Systems - Prentice Hall; 4 edition (March 10, 2014) - ISBN-10: 013359162X
[2] Douglas Comer - Operating System Design - The XINU Approach. CRC Press, 2015. ISBN : 9781498712439
[3] Silberschatz, Galvin, Gagne - Operating Systems Concepts - John Wiley & Sons; 10 edition (2018) – ISBN 978-1-119-32091-3

Software y Hardware

La versión de XINU que utilizamos es para arquitectura PC (x86). Ejecutamos el sistema operativo XINU en una máquina virtual llamada QEMU, que emula una PC básica.

El trabajo puede realizarse sobre las máquinas de los laboratorios (RECOMENDADO).

Quienes tengan Linux en sus casas, podrían intentar instalar todo lo necesario y llevarlo a cabo ahí también. Una tercera posibilidad es el acceso remoto RDP comentado en la web de la materia.

Ejercicio 1.

Estudie en XINU para que sirven las siguientes system calls (puede revisar el código fuente de XINU, o el libro XINU):

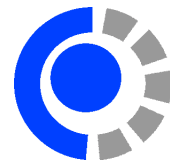
```
suspend(pid);  
resume(pid);  
getprio(pid);  
chprio(pid, newprio);  
getpid();  
sleepms(ms);
```

- a. ¿Se puede hacer un `suspend(pid)` del proceso `pid` estando el proceso `pid` en estado “sleeping”?
- b. Repase el TP anterior: ¿cuánto vale el quantum en XINU?. ¿Que problema o comportamiento puede haber si usted cambia el quantum a 100ms?
- c. Investigue cómo se llaman y cómo funcionan los algoritmos de planificación de procesos del sistema operativo Windows y Linux. Redacte una respuesta-aprendizaje. Esto implica: **no buscar en google/chatgpt y sólo copiar y pegar. Intente dar una explicación general.**

Ayuda: Abraham Silberschatz



Sistemas Operativos I
Trabajo Práctico Obligatorio 2
2025



Ejercicio 2. Estado sleeping and prioridades

- a. Desarrollar un programa en XINU que genere dos procesos para mostrar números primos. Un proceso debe mostrar los primos del 1 al 5000, y el segundo proceso del 5000 al 10000.
- b. Modifique el programa anterior para darle al primer proceso de cálculo de primos mayor prioridad que al segundo proceso. Verifique su ejecución e indique cómo se comporta el planificador de CPU de XINU al ejecutar estos dos procesos de este inciso.
- c. XINU provee el system call sleepms() para que un proceso voluntariamente “delegue” la CPU durante un tiempo en milisegundos. Al delegar la CPU, XINU coloca al proceso en estado “DURMIENDO” (estado “bloqueado” en la literatura teórica). Cuando el tiempo de dormir finaliza, XINU coloca al proceso en estado de “LISTO”.

Utilice este system call para modificar el comportamiento del programa del inciso b., de manera tal que el proceso 1, luego de 100 números primos calculados, duerma durante 10ms. Verifique su ejecución e indique cómo se comporta el planificador de CPU de XINU al ejecutar los dos procesos (siendo el proceso 1 de mayor prioridad que 2).

Ejercicio 3. Ejemplos de Planificación de CPU

Sea un sistema con la siguiente carga de procesos. El planificador de CPU del sistema es por prioridades, round-robin, apropiativo.

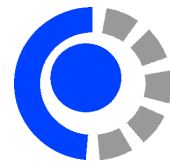
Proceso	Prioridad	Ráfaga	Arribo a la cola de listos
P1	8	15	0
P2	3	20	0
P3	4	20	20
P4	4	20	25
P5	5	5	45
P6	5	15	55

A más alto número de prioridad mayor prioridad. El quantum del sistema es de 10 unidades. A misma prioridad el planificador es round-robin.

- a. Mostrar el orden de ejecución y las ráfagas de CPU de los procesos.
- b. ¿Cuánto es el tiempo de turnaround para cada proceso? ¿Y el valor medio contando todos los procesos?
- c. ¿Cuál es el tiempo de espera para cada proceso? ¿Y el valor medio contando todos los procesos?



Sistemas Operativos I
Trabajo Práctico Obligatorio 2
2025



Ejercicio 4.

Sea un sistema con la siguiente carga de procesos.

Proceso	Ráfaga	Prioridad
P1	5	4
P2	3	1
P3	1	2
P4	7	2
P5	4	3

Los procesos arriban al sistema en el momento 0, en este orden: P1 , P2 , P3 , P4 , P5.

- Mostrar el orden de ejecución y las ráfagas de CPU de los procesos si el planificador es FCFS, SJF, con prioridades no-apropiativo, round-robin (quatum = 2).
- ¿Cuánto es el tiempo de turnaround para cada proceso? ¿Y el valor medio contando todos los procesos? (para cada algoritmo).
- ¿Cuál es el tiempo de espera para cada proceso? ¿Y el valor medio contando todos los procesos? (para cada algoritmo).
- ¿Cuál de los algoritmos produce el mínimo promedio de tiempo de espera?

Ejercicio 5.

- Averigue como puede ver el estado de cada proceso en ejecución de un sistema UNIX (Linux/MAC OS), etc. ¿Qué estados posibles existen en UNIX? ¿Qué significa el estado zombie en UNIX según Tanenbaum?
- Averigue como puede ver el estado de cada proceso en ejecución de un sistema UNIX (Linux/MAC OS), etc.