Practica N.4

Introduccion a los sistemas operativos

Guillermo Helfer

2022

PARTE 1

1. Un programa es estatico, no tiene PC y existe desde que se edita hasta que se borra.

Un proceso es un programa en ejecucion. Es dinamico, tiene PC, su ciclo de vida se comprende desde que se lo ejecuta hasta que se termina.

1. El tiempo de retorno es el tiempo que transcurre entre que el proceso llega al sistema hasta que completa su ejecucion.

El tiempo de espera es el tiempo en el que el proceso se encuentra en espera, es decir sin ejecutarse.

1. Promedio de ambos es un valor medio de tiempo que transcurre hasta que finaliza cualquiera de los dos.
2. Quantum (Q): medida que determina cuanto tiempo podria usar el procesador cada proceso (pequenio: overhead de context switch; grande: apropiacion del cpu)
3. apropiativo: el proceso en ejecucion puede ser interrumpido y llevado a la cola de listos (mayor overhead pero mejor servicio) (un proeso no monopoliza el procesador).

no-apropiativo: una vez que un proceso esta en estado de ejecucion, continua hasta que termina o se bloquea por algun evento.

1. **Short-term Scheduler**: decide a cual de los procesos en la cola de *ready* se elige para que use la CPU (que proceso se ejecutara)

**Medium-term Scheduler**: realiza el *swapping* entre el disco y la memoria cuando el SO lo determina (puede disminuir el grado de multiprogramacion)

**Long-term Scheduler**: determina que proceso pasara a ejecutarse.

1. **Dispatcher**: hace cambio de contexto, cambio de modo de ejecución...”despacha” el proceso elegido por el Short Term (es decir, “salta” a la instrucción a ejecutar).

PARTE 2:

1. **top:** provee una vista dinamica en tiempo real de un sistema de ejecucion.

**htop**: simil *top*, permite desplazarse vertical y horizontalmente e interacuar usando el mouse.

**ps**: muestra una captura de una seleccion de procesos.

**pstree**: simil *ps*, muestra en pantalla los procesos en forma de arbol

**kill**: envia la signal SIGTERM al proceso indicado con su PID

**pgrep**: mira entre los procesos que se estan ejecutando actualmente y lista sus PIDs.

**pkill**: enviara la signal especificada a cada proceso en lugar de listarlos en la stdout.

**killall**: envia la signal especificada a todos los procesos que se estan ejecutando segun los parametros especificados.

**renice**: altera la prioridad de la cola de tareas de uno o mas proceos que se estan ejecutando.

**xkill**: convierte el cursor en una *"X"* y con un click en cualquier ventana abierta mata el proceso.

**atop**: es un monitor interactivo que permite ver la carga de un sistema linux, muestra la ocupacion de los recursos de hardware mas criticos en un system level (desde un punto de vista de performance).

ej: cpu, memoria, disco, red.