

Procesos alternan (páganle a CPU y I/O)

CPU Bound

→ user + CPU q' S/O

I/O Bound

→ user + S/O.

Planificación: → elegir cuál proceso de los de la cola de turno va a ejecutarse.

Algoritmo de planificación: → cómo voy a ordenar la cola de turno.

→ No Apropiativo → Proceso se ejecuta hasta q' salen voluntariamente del CPU.

→ Se bloquea x e/s o término.

→ Interrupción x block → ^{finis} freno lo q' se está ejecutando y le pregunta a CPU si sigue o para.

(Probablemente sigue, no se romen devuena)

→ Apropiativo → Antes el proceso se ejecuta de CPU.

→ Cuando hay interrup. x block, Proceso se detiene; se pregunta o no el proceso o si para a otro.

Según el ambiente de trabajo necesitamos varios algoritmos con Finis.

→ Equidad: dar una porción de CPU a cada proceso.

→ Balance: Mantener ocupados todos los partes del sistema.

Procesos por lote/Batch → los q' no tienen interacción con user.

→ No hay user q' esperen una respuesta.

→ Algoritmo no apropiativo.

→ Maximizar Cost de Trabajo x Hora (Tiempo de espera)

Finis
valor

→ Minimizar tiempos el q' arranca y termina. (Tiempo de retorno)

FCFS → First come first served

STF → Shortest job first

(FIFO)

A B C D
8 4 4 4

el + corto primero.

B C D A
4 4 4 8

Algoritmo de prioridad

→ Colas múltiples

→ Varios colas según la prioridad de cada proceso y se ejecutan

primero los de +. Después, se hacen los de baja prioridad.

OPUS: Si hay un tiempo de espera se baja la prioridad.

→ Round robin

→ Round

Política de mecanismo.

→ usuario define política (q' cosa q' se desarrolla así)

→ q' algoritmo de planificación se aplica a lo q' se hace

→ kernel implementa mecanismo.

→ user/proceso/administrador con parámetros para determinar política.

interrupción x clock

↳ se para proceso y con él se g' hace

en NO aproximativo siempre sig. el proceso hasta q termina.
(pueden hacer cosas s/o en los puntos)