**ACTIVIDAD SOBRE** [**EL DESPACHADOR**](https://plataforma.utp.edu.co/mod/url/view.php?id=1568) **SEMANA 5**

1. **Qué objetivos debe cumplir un algoritmo de despacho.**

Los objetivos que debe cumplir un algoritmo de despacho son:

**Justicia:** Asegurarse que todos los procesos tengan su turno de CPU

**Eficiencia**: Mantener la CPU ocupada todo el tiempo

**Tiempo de respuesta:** Minimizar el tiempo de respuesta de los usuarios interactivos

**Rendimiento o productividad:** Maximizar el número de trabajos terminados por unidad de tiempo.

**Tiempo de espera:** Minimizar el tiempo medio de espera (En la cola READY) de los procesos.Tiempo transcurrido entre llegar y ser atendido.

1. **En qué consiste la planificación expropiadora y no expropiadora.**

* **Expropiadora:** Se realiza cuando en vez de liberar el microprocesador se elige un nuevo proceso a realizar.
* **No expropiadora:** Es cuando no se toma ninguna decisión de scheduling y se deja continuar ejecutando el mismo proceso.

1. **Cuales son los problemas del algoritmo FCFS.**

El algoritmo FCFS (First-come, First-served) no usa expropiación como tal, sino que su algoritmo consiste en atender los procesos en estricto orden de llegada a la cola READY la cual se maneja como una cola FIFO. El principal problema es que es el algoritmo es bastante malo. Teniendo en cuenta su funcionamiento , no es posible garantizar que los tiempos de espera sean los mínimos posibles, incluso podrían llegar a hacer los máximos.

En cuanto a la utilización de CPU se puede generar lo que es conocido como efecto convoy el cual se traduce en una baja utilización de esta u de los dispositivos I/O

1. **Cuales son los criterios para definir la prioridad en un proceso.**

Los criterios para definir la prioridad de un proceso son:

* Según categoría del usuario
* Según tipo de proceso: Sistema Interactivo, o por lotes; o bien, intensivo en CPU o intensivo I/O
* Según cuanto hayan ocupado la CPU hasta el momento.
* Para evitar que un proceso de baja prioridad sea postergado en demasía, aumentar prioridad mientras más tiempo lleva esperando: Envejecimiento
* Para evitar que un proceso de alta prioridad ejecute por demasiado tiempo, se le puede ir bajando la prioridad.

1. **En el algoritmo RR cuál es el problema de tener un Quantum muy grande.**

Si el quantum es muy grande, el algoritmo carrusel o Round - Robin se degenera en un FCFS

1. **En qué casos, la cpu es entregada al siguiente proceso de la cola ready en R-R.**

En el Round-Robin se le entrega la CPU por una tajada de tiempo predefinida, por lo que antes de ceder la CPU inicia un timer para que este provoque una interrupción cuando el tiempo asignado se acabe.

Por lo tanto el proceso se ejecuta hasta que se haga una llamada bloqueante o hasta que se use toda la tajada de tiempo, y cuando alguna de los dos casos ocurra, esta se entrega al siguiente en la cola READY.

1. **Entregue para el algoritmo anterior un ejemplo de algoritmo intraclase y uno interclase. hola**

**I**ntra-clase: Planificación

Inter-clase: Interactivos y los procesos por lotes

1. **En la planificación a dos niveles cuál debe ser el oficio del algoritmo a largo plazo..**

Decidir qué procesos han estado demasiado tiempo en memoria y deben pasar a disco para que dar oportunidad a otros procesos que han estado mucho rato en el disco.

1. **Explique algoritmos para múltiples procesadores.**

Los algoritmos que usan planificación en multiprocesadores corren riesgos de que la carga entre procesadores sea desbalanceada ya que unos pueden tener una cola muy larga de procesos para ejecutar mientras otros están desocupados para mitigar esto se realiza lo siguiente:

* Cada procesador es responsable de su propia planificación y puede ir sacando procesos de la cola para ejecutar pero con la falta de sincronización para sacarlos y es bastante ineficiente
* Usar el multiprocesador asimétrico donde solo un procesador se encarga de la planificación y de qué procesos ejecutan los demás.

1. **Qué métodos se pueden usar para evaluar algoritmos.**

* **Modelación determinista:** Se toma un caso en particular y se toma analiticamente la medda de interes.
* **Modelos de colas:** La idea es determinar experimentalmente una distribución de la duraciones de las fases de CPU e I/O, de la misma forma determinar la distribución probabilística de los tiempos de llegada de los nuevos procesos, conociendo esto se pueden determinar métricas como utilización (PORCENTAJE DE USO DE LOS SERVIDORES), tamaños medios de las colas y tiempos medios de espera.
* **Simulaciones:** En vez de resolver el modelo matemáticamente la idea es programarlo y ver cómo funciona, el simulador tiene una variable que representa el reloj y a medida que esta se incrementa se modifica el estado del simulador para reflejar las actividades de los diferentes elementos simulados.
* **Implementación:** Lo mejor es programar el sistema operativo con el algoritmo que queremos evaluar, echarlo a andar y ver cómo se comporta.