




Gestión de procesos

- 
1. Proceso y PCB
 2. Estados de los procesos
 3. Planificación de procesos
 4. Algoritmos de planificación de procesos

1. Proceso y PCB

- Un **proceso** es programa en ejecución
- Cuando se ejecuta un proceso, éste necesita que el SO le suministre una serie de recursos: memoria, impresora, scanner, ficheros...
- Si se ejecutan varios procesos de forma concurrente, lo cual ocurre normalmente, todos ellos necesitan recursos para poderse ejecutar

1. Proceso y PCB

- Cada proceso tendrá una estructura de datos llamada bloque de control de proceso
- Esta estructura identifica al proceso de forma única y sirve para controlar su correcta ejecución

1. Proceso y PCB

- Cada **BCP** contiene la siguiente información:
 - **Estado** actual del proceso
 - Identificador del proceso (**PID**). Un numero que asigna el SO.
 - **Prioridad** del proceso. Asignada de forma automática por el SO. Puede ser modificada por el administrador del sistema.

1. Proceso y PCB

- **Ubicación en memoria.** Relación de las zonas de memoria donde el SO ha ubicado el proceso.
- **Recursos** utilizados . Relación de los recursos hw y sw que el proceso ha de tener asignados para poder ejecutarse.

La asignación de estos recursos en el momento en que el proceso comience a ejecutarse la realizará el SO.

2.Estados de los procesos

■ **Estado** de un proceso: Posición que ocupa en cuanto al uso de recursos hardware del equipo

Creación: Se crea la estructura de datos que se utilizan en la administración de dicho proceso

Preparado: Se encuentra un proceso que necesita la CPU para ejecutarse.

En ejecución: Cuando el proceso posee la CPU

En espera: El proceso espera que se produzca algún suceso, como por ejemplo una operación solicitada de E/S.

Terminado: El proceso acaba su ejecución.

3. Planificación de procesos

- En multiprogramación, múltiples procesos son mantenidos en memoria principal.
- La idea es tener algún proceso ejecutándose en todo momento con la finalidad de maximizar el uso del CPU.
- El Tiempo Compartido fue concebido con la idea de conmutar o cambiar continuamente la CPU entre procesos de forma tal que los usuarios puedan interactuar con sus programas mientras se están ejecutando.

3. Planificación de procesos

- Cuando un proceso entra al sistema es colocado en una cola de trabajos. Una vez que el proceso se encuentra en memoria principal y está listo para ejecutar, este es colocado en la cola de procesos preparados.
- Cuando al proceso se le asigna la CPU, ejecuta por un tiempo y eventualmente el proceso terminará, o será interrumpido o esperará por la ocurrencia de algún evento.
- En un sistema de Tiempo Compartido, el SO debe decidir qué proceso debe ejecutarse a continuación de los que hay en la cola de preparados.

3. Planificación de procesos

¿Como funciona la gestión de procesos en un sistema multihilo?

- Tenemos abierto un documento Word, sin cerrarlo, abrimos otro documento Word
- El proceso Word, tendrá dos hilos, hebras o threads diferentes, pero el proceso Word sigue siendo el mismo, se está ejecutando una sola vez
- Cada hilo no es un proceso propiamente dicho, sin embargo aparecen en la planificación de procesos como si lo fueran
- Para nuestro SO tendremos un solo proceso con diversas tareas a ejecutar. El planificador se encarga de que cuando le toque ejecutarse a nuestro proceso, las tareas se vayan alternando, con la diferencia de que al ser un mismo proceso, no se libera espacio de memoria ni procesador, como se haría en el caso de alternar procesos.

3. Planificación de procesos

- Todos los hilos de un proceso comparten la misma zona de memoria,
- De esta forma, la comunicación entre los hilos de un proceso es más rápida y eficiente que si los hilos fuesen procesos diferentes
- Cuando un hilo se ejecuta, el resto están en espera o bloqueados.
- Habrá un solo hilo ejecutándose



3. Planificación de procesos

- La técnica de planificación permite indicar al ordenador los procesos que deben ejecutarse y los estados que deben adoptar.
- Mediante los algoritmos de planificación se consigue decidir que proceso debe ejecutarse.



4. Algoritmos de Planificación

Se denomina planificador a la parte del sistema operativo encargada de determinar que proceso pasará al estado activo entre todos los procesos preparados.

Entre los criterios que existen para compararlos están:

Tiempo de espera

Tiempo de retorno

Tiempo de respuesta

4. Algoritmos de planificación

- **Tiempo de espera.** Hace referencia a la suma de los tiempos que se pasa el proceso en la cola de preparados.
- **Tiempo de retorno.** Es el tiempo que va a tardar un proceso en ejecutarse completamente.
- **Tiempo de respuesta.** Es el tiempo que transcurre desde que se hace una petición al sistema y este empieza a responder.

4. Algoritmos de planificación

■ Algoritmo FCFS

- Los ciclos de CPU asignados a cada proceso se asignan en función de una cola FIFO.
- Al primer proceso de la cola se le asigna tiempos o ciclos de CPU hasta que termina completamente



4. Algoritmos de planificación

- Algoritmo SJF

Concede la CPU al proceso que menos tiempo necesita la CPU de los que hay en la cola de preparados.

4. Algoritmos de planificación

■ Algoritmo de rueda o Round-Robin

- Asigna rotativamente tiempos de ejecución a los procesos, según una cola FIFO
- El tiempo de ejecución es el mismo para todos los procesos (mismo quantum)
- Se asigna cpu al proceso que se encuentra en la cabeza de la cola.
- Cuando un proceso termina su quantum, se coloca el último de la cola
- Los nuevos procesos son agregados al final de la cola

4. Algoritmos de planificación


■ Por Prioridades:

- ☐ Se asigna una prioridad a cada proceso y se otorga el uso del procesador al que la tenga mayor
- ☐ Habrá una cola de procesos para cada prioridad.
- ☐ Cada cola de procesos se gestionará con otro algoritmo de planificación
- ☐ Si le modificamos a un proceso la prioridad, cambiará de cola

4. Algoritmos de planificación

Múltiples colas

- Es una combinación de los anteriores y es el implementado en los sistemas operativos actuales.
- En términos generales, los procesos de una misma prioridad estarán en la misma cola. Cada cola será gestionada con otro algoritmo.
- Los procesos de colas de inferior prioridad no pueden ejecutarse hasta que no se hayan vaciado las colas de procesos de mayor prioridad.

- 
- **Planificación con colas de múltiples niveles y realimentación: „**
 - Se permite que un proceso pase de una cola a otra.
 - „Si un proceso gasta demasiado tiempo de CPU, se le pasa a una cola de menor prioridad. „
 - Si un proceso espera mucho tiempo en una cola de baja prioridad se le puede pasar a una de mayor prioridad, evitando la inanición.