

Ciclo de Vida de un Proceso y Gestión de Memoria

Un proceso pasa por varios estados a lo largo de su ciclo de vida. Es importante comprender cómo cada estado se relaciona con la gestión de memoria para entender cómo el sistema operativo organiza y administra los recursos de la máquina.

1. ****Nuevo / Listo****

- El sistema operativo reserva memoria para el código, datos y pila del proceso.
- Se crea la estructura ****PCB (Process Control Block)**** que contiene el estado y la información de memoria del proceso.

2. ****Ejecución****

- El proceso usa la CPU y accede a su espacio de memoria.
- Puede solicitar más memoria usando llamadas del sistema (por ejemplo, ``malloc()``, ``new``).
- El sistema operativo protege la memoria del proceso para evitar que otros procesos la accedan.

3. ****Bloqueado****

- El proceso sigue utilizando su espacio de memoria.
- No puede liberar memoria porque el proceso no ha finalizado.
 - Si hay falta de RAM, el sistema operativo puede mover el proceso a memoria secundaria (swap).

4. ****Finalizado****

- El proceso libera toda la memoria que estaba usando.
- El sistema operativo borra la estructura PCB y devuelve la memoria al sistema.

Técnicas de Gestión de Memoria:

- **Asignación continua**: Asignación de un bloque de memoria contiguo (prácticamente en desuso).
- **Segmentación**: Dividir la memoria en segmentos lógicos como código, datos y pila.
- **Paginación**: Dividir la memoria en pequeñas páginas para optimizar el uso de RAM.
- **Memoria Virtual**: Permite que un proceso use más memoria de la que realmente tiene, haciendo uso de swap.

Ejemplo de Paginación:

Si un proceso requiere 20 KB de memoria y el sistema usa páginas de 4 KB, el sistema le asignará 5 páginas ($5 * 4 \text{ KB}$). Las páginas pueden estar distribuidas en diferentes partes de la memoria física, y el proceso usa direcciones virtuales que el sistema traduce a direcciones físicas.