

Administración de memoria

3.1 Política y filosofía

1. ¿Cuál es la diferencia entre fragmentación interna y externa?

La **fragmentación interna** ocurre cuando un bloque de memoria es más grande que el tamaño solicitado y se desperdicia el exceso de espacio.

Por ejemplo: digamos que un asignador solo puede proporcionar memoria en fragmentos de 4 bytes. Si un programa solicita 29 bytes de memoria, en realidad recibirá una porción de 32 bytes. Dado que el programa utiliza sólo 29 de los 32 bytes disponibles, los 3 bytes sobrantes de memoria se desperdician.

Esta memoria que no se usa queda inservible para otros procesos y limita la cantidad que se pueden ejecutar al mismo tiempo. Ya que reduce la memoria funcional óptima. Esto puede aumentar el tiempo de procesamiento ya que debe buscar un bloque adecuado para reducir la fragmentación.

La **fragmentación externa** ocurre cuando hay espacios entre los bloques asignados y la memoria no se puede usar para solicitudes más grandes.

Por ejemplo: digamos que el asignador asigna varios bloques de memoria en una línea continua. Si uno de los bloques intermedios se libera porque el proceso que lo estaba usando dejó de ejecutarse, el bloque libre se fragmenta.

Esto hace que solo se pueda usar ese bloque aislado solo si hay una solicitud que quepa dentro, pero no puede usarse para solicitudes de memoria mayores. Y no puede unirse como los bloques libres.

Entonces la **fragmentación interna** ocurre dentro del espacio del bloque que se solicita, mientras que la **fragmentación externa** ocurre en los bloques contiguos o espacios entre los bloques.

2. Investiga y explica las políticas de reemplazo de páginas en sistemas operativos. ¿Cuál consideras más eficiente y por qué?

Las **políticas de reemplazo de páginas** son algoritmos utilizados en sistemas operativos para decidir qué página en la memoria debe ser reemplazada cuando se necesita liberar espacio para cargar una nueva página.

Una página es una unidad de memoria fija en la que se divide el espacio de direcciones de un proceso. Cada página puede tener un tamaño de 4 KB, 8 KB o más, dependiendo de la arquitectura del sistema.

La memoria física (RAM) se divide en bloques de tamaño fijo llamados marcos de página, que son del mismo tamaño que las páginas.

Hay varios algoritmos de reemplazo de páginas, cada uno con sus propias características y estrategias:

1. FIFO (First-In, First-Out): Este algoritmo reemplaza la página más antigua en la memoria. Se usa una cola, al cargar una página nueva se ingresa en el último lugar. Aunque las colas FIFO son simples e intuitivas, no se comportan de manera aceptable en la aplicación práctica, por lo que es raro su uso en su forma simple.

Una desventaja es la anomalía Belady, consiste en que podemos quitar de memoria una página de memoria muy usada, solo porque es la más antigua.

2. LRU (Least Recently Used): Reemplaza la página que no ha sido utilizada durante el periodo más largo de tiempo. Este algoritmo utiliza la idea de que las páginas que han sido accedidas recientemente tienen más probabilidad de ser accedidas nuevamente en el futuro.

La desventaja es que gasta recursos para rastrear el orden de uso de las páginas.

3. Optimal: Este es un algoritmo teórico que reemplaza la página que no será utilizada por el mayor periodo de tiempo en el futuro. Para esto el sistema operativo debería ver en cuánto tiempo será usada cada página en memoria y elegir la que está más distante.

El problema de este método es que necesita conocimiento del futuro, por lo que es difícil de implementar.

4. Algoritmo de reloj: Es una variación de FIFO que utiliza un "bit de referencia" para decidir qué páginas reemplazar. En caso de estar fijado (en 1) se cambia el bit a 0 y se lo coloca al final de la obstrucción, actualizando su tiempo de carga como si recién hubiera llegado al procesador. De esta forma, se le da una segunda oportunidad. Si el bit se encuentra sin fijar(en 0), la página se saca de memoria.

La desventaja es que en algunos casos suele ser menos precisa que LRU.

5. NRU (Not Recently Used): Cuando una página es referenciada, fija el bit de referencia para esa página. Similarmente, cuando una página es modificada, fija su bit de modificación.

Cuando una página debe ser reemplazada, el sistema operativo divide las páginas en cuatro categorías:

- Categoría 0: No referenciada, No modificada

- Categoría 1: No referenciada, modificada
- Categoría 2: referenciada, No modificada
- Categoría 3: referenciada, modificada

Las mejores páginas para cambiar son las que se encuentran en la categoría 0, mientras que las peores son las de la categoría 3.

En algunos casos es más óptimo LRU o reloj.

El más **eficiente** podría ser el optimal, pero solo de forma teórica ya que es el más difícil de llevarlo a la practica, así que yo elegiría LRU como el más eficiente, ya que aunque puede gastar más recursos que otros algoritmos, es el que mejor referencia el estado de las páginas en los sistemas para ser reemplazados.