*017*

*2*



# Trabajo Práctico Nº 5

Administración de la Memoria Principal

**1.-** Explique a que hacen referencia los siguientes términos:

* Dirección Lógica o Virtual
* Dirección Física

Dirección Lógica: • Es una dirección que enmascara o abstrae una dirección física

• Referencia a una localidad en memoria

• Se la debe traducir a una dirección física

Dirección Física: • Es la dirección real. Es con la que se accede efectivamente a memoria

• Representa la dirección absoluta en memoria principal

**2.-** En la técnica de Particiones Múltiples, la memoria es divida en varias particiones y los procesos son ubicados en estas, siempre que el tamaño del mismo sea menor o igual que el tamaño de la partición.

Al trabajar con particiones se pueden considerar 2 métodos (independientes entre si):

* + Particiones Fijas
  + Particiones Dinámicas

1. Explique como trabajan estos 2 métodos. Cite diferencias, ventajas y desventajas.
2. ¿Qué información debe disponer el SO para poder administrar la memoria con estos métodos?
3. Realice un gráfico indicado como realiza el SO la transformación de direcciones lógicas a direcciones físicas.

a) **Particiones Fijas:** En este método, la memoria se divide en particiones de tamaño fijo antes de que los procesos comiencen a ejecutarse. Cada partición puede contener un solo proceso, y si un proceso no cabe en ninguna partición disponible, debe esperar hasta que haya una partición lo suficientemente grande para él. Algunas de las características y diferencias clave son:

* **Ventajas:**
  + **Simplicidad:** Es más fácil de implementar y comprender, ya que el sistema operativo solo necesita gestionar particiones de tamaño fijo.
* **Desventajas:**
  + **Fragmentación Externa:** Puede haber fragmentación externa, donde la memoria está dividida en varias particiones, pero no se pueden utilizar para ejecutar procesos debido a su tamaño.
  + **Ineficiencia en el uso de la memoria:** Puede haber procesos más pequeños que ocupen una partición completa, lo que lleva a un uso ineficiente de la memoria.

**Particiones Dinámicas:** En este método, las particiones no tienen un tamaño fijo y se crean dinámicamente según sea necesario para acomodar los procesos. A medida que los procesos entran y salen, las particiones se ajustan para evitar fragmentación externa. Algunas características y diferencias clave incluyen:

* **Ventajas:**
  + **Mejor utilización de la memoria:** Al ser dinámico, se adapta mejor a procesos de diferentes tamaños, evitando asignar particiones más grandes de lo necesario.
* **Desventajas:**
  + **Complejidad:** La implementación es más compleja, ya que el sistema operativo debe gestionar de manera eficiente la creación y liberación dinámica de particiones.
  + **Posible fragmentación interna:** Puede haber fragmentación interna si un proceso no ocupa toda la partición asignada.

b) **Información necesaria para administrar la memoria:** Para administrar la memoria con estos métodos, el sistema operativo debe tener información como:

* **Para Particiones Fijas:**
  + Tamaño de cada partición.
  + Estado de ocupación de cada partición (si está ocupada por un proceso o no).
* **Para Particiones Dinámicas:**
  + Tamaño y ubicación de cada partición actual.
  + Información sobre procesos en ejecución y su tamaño.
  + Algoritmo de asignación de particiones

Diagrama

Descripción generada automáticamentePrincipio del formulario

c)

**3.-** Al trabajar con particiones fijas, los tamaños de las mismas se pueden considerar:

* + Particiones de igual tamaño.
  + Particiones de diferente tamaño. Cite ventajas y desventajas de estos 2 métodos.

**Particiones de Igual Tamaño:**

* **Ventajas:**
  + **Simplicidad:** Es más fácil de implementar y entender porque todas las particiones tienen el mismo tamaño.
  + **Mayor fragmentacion interna:** La probabilidad de fragmentación interna es menor ya que los procesos no se ajustan perfectamente al tamaño de la partición.
* **Desventajas:**
  + **Uso ineficiente de la memoria:** Puede llevar a un uso ineficiente de la memoria, ya que los procesos deben ajustarse al tamaño fijo de la partición, incluso si son más pequeños.

**Particiones de Diferente Tamaño:**

* **Ventajas:**
  + **Mejor adaptabilidad:** Permite adaptarse mejor a procesos de diferentes tamaños, asignando particiones más ajustadas a los procesos.
  + **Mayor eficiencia en el uso de la memoria:** Puede reducir la fragmentación interna al asignar particiones más ajustadas, mejorando la eficiencia general de la memoria.
* **Desventajas:**
  + **Mayor complejidad:** La implementación es más compleja ya que el sistema operativo debe gestionar particiones de diferentes tamaños.
  + **Posible fragmentación externa:** Aunque se reduce, aún puede haber fragmentación externa si las particiones no se gestionan adecuadamente.

**4.-** Fragmentación

Ambos métodos de particiones presentan el problema de la fragmentación:

* + Fragmentación Interna (Para el caso de Particiones Fijas)
  + Fragmentación Externa (Para el caso de Particiones Dinámicas)

1. Explique a que hacen referencia estos 2 problemas
2. El problema de la Fragmentación Externa es posible de subsanar. Explique una técnica que evite este problema.

La fragmentación se produce cuando una localidad de memoria no puede ser utilizada por no encontrarse en forma contigua

1. Fragmentación Interna: • Se produce en el esquema de particiones fijas, por ejemplo

• Es interna a la localidad asignada

• Es la porción de la localidad que queda sin utilizar

Fragmentación Externa: • Se produce en particiones dinámicas, por ejemplo

• Son huecos que van quedando en la memoria a medida que . los procesos finalizan

• Al no encontrarse en forma contigua puede darse el caso de . que tengamos memoria libre para alocar un proceso, pero . . que no la podamos utilizar

1. Solución → compactación → muy costosa

**5.-** Paginación

1. Explique como trabaja este método de asignación de memoria.
2. ¿Qué estructuras adicionales debe poseer el SO para llevar a cabo su implementación?
3. Explique, utilizando gráficos, como son transformadas las direcciones lógicas en físicas.
4. En este esquema: ¿Se puede producir fragmentación (interna y/o externa)?
5. • La memoria se divide en porciones de igual tamaño llamadas marco

• El espacio de direcciones de los procesos se divide en porciones de igual tamaño denominadas páginas

• Tamaño página = tamaño marco = 512 bytes (generalmente)

1. El SO mantiene una tabla de páginas para cada proceso, la cual contiene el marco donde se . encuentra cada página y un registro de control
2. d) No es posible que ocurra fragmentación externa (Todas las paginas son iguales), pero si

interna

Gráfico

Descripción generada automáticamenteTabla

Descripción generada automáticamented)

**6.-** Cite similitudes y diferencias entre la técnica de paginación y la de particiones fijas.

**Similitudes entre Paginación y Particiones Fijas:**

1. Ambas dividen la memoria en unidades fijas.
2. Se pueden implementar esquemas de protección de memoria en ambas.
3. El sistema operativo controla la asignación y gestión de la memoria en ambos casos.
4. Puede ocurrir fragmentación interna, y no ocurre fragmentación externa (Particiones fijas idénticas)

**Diferencias entre Paginación y Particiones Fijas:**

1. **Flexibilidad en Tamaño:**
   * *Particiones Fijas:* Tamaño predeterminado fijo (Pero no necesariamente iguales).
   * *Paginación:* Permite tamaños fijos e iguales de asignación.
2. **Implementación de Tablas:**
   * *Particiones Fijas:* No requiere tablas de asignación.
   * *Paginación:* Requiere tablas de páginas para mapear direcciones.
3. **Reubicación:**
   * *Particiones Fijas:* Sin reubicación, direcciones fijas.
   * *Paginación:* Permite reubicación más flexible.

**7.-** Suponga un sistema donde la memoria es administrada mediante la técnica de paginación, y donde:

* Imagen de la pantalla de un celular con letras

  Descripción generada automáticamente con confianza bajaEl tamaño de la página es de 512 bytes
* Cada dirección de memoria referencia 1 byte.
* Los marcos en memoria principal de encuentran desde la dirección física 0. Suponga además un proceso con un tamaño 2000 bytes y con la siguiente tabla de paginas:

1. Realice los gráficos necesarios (de la memoria, proceso y tabla de paginas) en el que reflejen el estado descrito.
2. Indicar si las siguientes direcciones lógicas son correctas y en caso afirmativo indicar la dirección física a la que corresponden:

i) 35

* 1. 512
  2. 2051
  3. 0

v) 1325

vi) 602

1. Indicar, en caso de ser posible, las direcciones lógicas del proceso que se corresponden si las siguientes direcciones físicas:

i) 509

ii) 1500

iii) 0

iv) 3215

v) 1024

vi) 2000

1. ¿Indique, en caso que se produzca, la fragmentación (interna y/o externa)?

*2017*

Diagrama

Descripción generada automáticamente

2. Usando las tablas anteriormente descriptas

Nro pagina = N= Direccion / Tamaño marco

Desplazamiento = D = Direccion % Tamaño marco

Direccion Física = Nro Marco donde se guarda la pagina \* Tamaño marco + Desp.

Por lo tanto queda:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tamaño marco = 512 bytes | | | | | |
| Ejercicio | Direccion Logica | Nro Pagina | Nro Marco | Desplazamiento | Direccion Fisica |
| I | 35 | 0 | 3 | 35 | 1571 |
| II | 512 | 1 | 5 | 0 | 2560 |
| III | 2051 | Excedido (Dir max=2k), avisando S.O | | | |
| IV | 0 | 0 | 3 | 0 | 1536 |
| V | 1325 | 2 | 2 | 275 | 1299 |
| VI | 602 | 1 | 5 | 90 | 2650 |

1. Usando las tablas anteriormente descriptas

Nro marco = N= Direccion / Tamaño marco

Desplazamiento = D = Direccion % Tamaño marco

Direccion Logica = Nro pagina que coniene el marco \* Tamaño marco + Desp.

Por lo tanto queda

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tamaño marco = 512 bytes | | | | | |
| Ejercicio | Direccion Fisica | Nro Marco | Nro Pagina | Desplazamiento | Direccion Logica |
| I | 509 | 0 | No corresponde (Marco 0 otro programa), Avisando S.O | | |
| II | 1500 | 2 | 2 | 476 | 1500 |
| III | 0 | 0 | No corresponde (Marco 0 otro programa), Avisando S.O | | |
| IV | 3215 | 6 | 3 | 143 | 1679 |
| V | 1024 | 2 | 2 | 0 | 1024 |
| VI | 2000 | 3 | 0 | 464 | 464 |

1. Ocurre
2. fragmentación interna

**8.-** Considere un espacio lógico de 8 paginas de 1024 bytes cada una, mapeadas en una memoria física de 32 marcos. (Cada dirección de memoria referencia 1 byte)

1. ¿Cuantos bits son necesarios para representar una dirección lógica?
2. ¿Cuantos bits son necesarios para representar una dirección física?

a) Para representar una dirección lógica en un espacio lógico de 8 páginas de 1024 bytes cada una, se necesitan log⁡2(8\*1024)=log2​(8,192)=13 bits.

b) Para representar una dirección física en una memoria física de 32 marcos, donde cada marco es de 1024 bytes, se necesitan log⁡2(32\*1024)=log2​(32,768)=15 bits.

**9.-** Segmentación

1. Explique como trabaja este método de asignación de memoria.
2. ¿Qué estructuras adicionales debe poseer el SO para llevar a cabo su implementación?
3. Explique, utilizando gráficos, como son transformadas las direcciones lógicas en físicas.
4. En este esquema: ¿Se puede producir fragmentación (interna y/o externa)?

**a) Funcionamiento:** En la técnica de segmentación, la memoria se divide en segmentos lógicos que representan unidades lógicas o funcionales del programa, como el código, los datos y la pila. Cada segmento tiene un tamaño variable y está asociado con un número que indica su función. Los procesos se dividen en segmentos, y cada segmento puede crecer o decrecer de manera independiente según las necesidades del programa.

Diagrama

Descripción generada automáticamente**b) Estructuras Adicionales:** Para implementar la segmentación, el sistema operativo debe mantener una **Tabla de Segmentos:** Es una tabla que mapea los identificadores de segmentos a sus ubicaciones en memoria física. Cada entrada de la tabla de segmentos contiene la base y el límite del segmento.

c) La cpu se fija que la dirección este en un segmento del programa, luego, usando un registro base y limite, se fije que la dirección que tiene efectivamente este contenida en el segmento

1. Se puede producir fragmentación externa

**10.-** Cite similitudes y diferencias entre la técnica de segmentación y la de particiones dinámicas.

**Similitudes entre Segmentación y Particiones Dinámicas:**

1. **División de Memoria:** Ambas técnicas dividen la memoria en unidades más pequeñas para facilitar la gestión y asignación de recursos a los procesos.
2. **Adaptabilidad:** Tanto la segmentación como las particiones dinámicas permiten a los procesos crecer o decrecer en tamaño durante su ejecución, adaptándose a las necesidades cambiantes.
3. **Fragmentación Externa:** Ambas pueden experimentar fragmentación externa

**Diferencias entre Segmentación y Particiones Dinámicas:**

1. **Unidad Básica:**
   * *Segmentación:* Divide la memoria en segmentos lógicos de diferentes tamaños según las necesidades del proceso.
   * *Particiones Dinámicas:* Divide la memoria en bloques de tamaño variable según la solicitud del proceso.
2. **Tamaño de las Unidades:**
   * *Segmentación:* Los segmentos pueden tener tamaños variados y contener información lógica relacionada.
   * *Particiones Dinámicas:* Las particiones pueden tener tamaños variables, pero no tienen una relación lógica específica entre ellas.
3. **Gestión de la Memoria:**
   * *Segmentación:* Cada segmento puede crecer o decrecer dinámicamente.
   * *Particiones Dinámicas:* Las particiones pueden ajustarse dinámicamente para adaptarse a los procesos.
4. **Implementación:**
   * *Segmentación:* Requiere una tabla de segmentos para mantener la correspondencia entre las direcciones lógicas y físicas.
   * *Particiones Dinámicas:* Requiere una estructura de datos para mantener el estado de las particiones y sus asignaciones.

**11.-** Cite similitudes y diferencias entre la técnica de paginación y segmentación.

**Similitudes entre Paginación y Segmentación:**

1. **División de Memoria:** Ambas técnicas dividen la memoria para facilitar la gestión de procesos y mejorar la eficiencia en el uso de recursos.
2. **Protección de Memoria:** Tanto en paginación como en segmentación, se pueden implementar esquemas de protección de memoria para evitar accesos no autorizados.
3. **Adaptabilidad:** Ambas permiten que los procesos crezcan o disminuyan en tamaño durante su ejecución, brindando flexibilidad en la gestión de la memoria.

**Diferencias entre Paginación y Segmentación:**

1. **Unidad Básica:**
   * *Paginación:* Divide la memoria en bloques de tamaño fijo llamados "páginas".
   * *Segmentación:* Divide la memoria en segmentos lógicos de tamaño variable según la naturaleza del contenido.
2. **Tamaño de las Unidades:**
   * *Paginación:* Todas las páginas tienen el mismo tamaño, definido a nivel del sistema.
   * *Segmentación:* Los segmentos pueden tener tamaños variables y representan unidades lógicas relacionadas (como código, datos, pila).
3. **Fragmentación Interna:**
   * *Paginación:* No hay fragmentación interna, ya que las páginas son de tamaño fijo y los procesos se ajustan a múltiplos enteros de páginas.
   * *Segmentación:* Puede haber fragmentación interna, ya que los segmentos pueden tener tamaños variables y los procesos no siempre llenan completamente un segmento.
4. **Fragmentación Externa:**
   * *Paginación:* No hay fragmentación externa, ya que las páginas se asignan de manera independiente.
   * *Segmentación:* Puede haber fragmentación externa si los segmentos no están contiguos en memoria.
5. **Implementación:**
   * *Paginación:* Requiere tablas de páginas para mapear direcciones lógicas a direcciones físicas.
   * *Segmentación:* Requiere tablas de segmentos para realizar el mapeo entre direcciones lógicas y físicas.
6. **Relación Lógica:**
   * *Paginación:* No implica una relación lógica específica entre las páginas.
   * *Segmentación:* Los segmentos tienen una relación lógica, representando diferentes partes funcionales de un programa.

**12.-** Dado un S.O. que administra la memoria por medio de segmentación paginada, y teniéndose disponibles las siguientes tablas:

# Tabla de Segmentos Tabla de Paginas

|  |  |
| --- | --- |
| Núm. Seg. | Dir. base |
| 1 | 500 |
| 2 | 1500 |
| 3 | 5000 |

|  |  |  |
| --- | --- | --- |
| Nro. Segmento | Nro. Pagina | Direc. Base |
| 1 | 1 | 40 |
| 2 | 80 |
| 3 | 60 |
| 2 | 1 | 20 |
| 2 | 25 |
| 3 | 0 |
| 3 | 1 | 120 |
| 2 | 150 |

Indicar las direcciones físicas correspondientes a las siguientes direcciones lógicas (segmento,pagina,desplazamiento):

i) (2,1,1) 1500 + 20 + 1 = 1521

ii) (1,3,15) 500 + 60 + 15 =575

iii) (3,1,10) 5000 + 120 + 10 = 5130

iv) (2,3,5) 1500 + 0 + 5 =1505

**13.-** Memoria Virtual

1. Describa que beneficios introduce este esquema de administración de la memoria.
2. ¿En que se debe apoyar el SO para su implementación?
3. Al implementar está técnica utilizando paginación por demanda, las tablas de paginas de un proceso deben contar con información adicional además del marco donde se encuentra la página. ¿Cuál es está información? ¿ Porque es necesaria?
4. Permite tener mas páginas que las que entran en la R.A.M (Ejecutar procesos mas grandes), mejorando la multiprogramación y la administración de memoria.
5. Para implementar la memoria virtual, se necesitan estas funcionalidades:

**Manejo de Paginación o Segmentación:** El sistema operativo debe admitir técnicas de paginación o segmentación para gestionar las transferencias de datos entre la memoria principal y el almacenamiento secundario (disco).

**Algoritmos de Reemplazo de Página:** Para gestionar la carga y descarga de páginas entre la memoria principal y el almacenamiento secundario, se necesitan algoritmos eficientes de reemplazo de página.

**Gestión de Espacio en Disco:** El sistema operativo debe gestionar el espacio en disco para almacenar las partes no residentes de los procesos cuando no están en memoria principal.

**Gestión de Fallos de Página:** Debe manejar los fallos de página, es decir, la situación en la que una página necesaria para la ejecución de un proceso no se encuentra en la memoria principal.

1. Cuando se implementa la técnica de paginación por demanda en memoria virtual, las tablas de páginas de un proceso deben contener información adicional además del marco donde se encuentra la página. La información adicional incluye:

**Bit de Presente/Ausente:** Indica si la página está actualmente en la memoria principal

**Bit de Modificado/No Modificado:** Indica si la página ha sido modificada desde que fue cargada en la memoria principal

**Bits de Protección:** Permiten establecer permisos de acceso a la página, como lectura, escritura o ejecución.

*2017*

**14.-** Fallos de Página (Page Faults):

1. ¿Cuándo se producen?
2. ¿Quién es responsable de detectar un fallo de página?
3. Describa las acciones que emprende el SO cando se produce un fallo de página.
4. Los fallos de página ocurren cuando un programa intenta acceder a una página que no está en la memoria principal.
5. El hardware detecta los fallo de página. Cuando se intenta acceder a una página que no está en la memoria, se genera una interrupción conocida como un fallo de página. El sistema operativo es responsable de manejar esta interrupción y tomar las acciones necesarias para cargar la página requerida en la memoria.
6. Verificación de la Página Ausente: El sistema operativo verifica la tabla de páginas para determinar si la página solicitada está presente en la memoria principal.

Carga de la Página desde el Almacenamiento Secundario: Si la página no está presente, el sistema operativo carga la página desde el almacenamiento secundario (generalmente un disco) a la memoria principal.

Actualización de las Tablas de Páginas: Se actualizan las tablas de páginas para reflejar que la página ahora está en la memoria y se actualizan los bits de control, como el bit de presente.

Además, la instrucción que causó el fallo de página se reinicia, permitiendo que el programa continúe su ejecución normalmente ahora que la página necesaria está en la memoria.

**15.-** Direcciones:

1. Si se dispone de una espacio de direcciones virtuales de 32 bits, donde cada dirección referencia 1 byte:
   1. ¿Cuál es el tamaño máximo de un proceso (recordar “espacio virtual”)?
   2. Si el tamaño de pagina es de 512Kb. ¿Cuál es el número máximo de paginas que puede tener un proceso?
   3. Si el tamaño de pagina es de 512Kb. y se disponen de 256 Mb. de memoria real ¿Cuál es el número de marcos que puede haber?
   4. Si se utilizaran 2 Kb. para cada entrada en la tabla de páginas de un proceso: ¿Cuál seria el tamaño máximo de la tabla de páginas de cada proceso?
2. **Tamaño Máximo de un Proceso:** El tamaño máximo de un proceso se determina por el espacio de direcciones virtuales. Dado que se dispone de un espacio de direcciones virtuales de 32 bits, donde cada dirección referencia 1 byte, el tamaño máximo de un proceso sería de 2^32 bytes, que es igual a 4 GB.
3. Si el tamaño de página es de 512 KB, el número máximo de páginas que puede tener un proceso se calcula dividiendo el tamaño máximo del proceso entre el tamaño de la página:

Imagen que contiene Tabla

Descripción generada automáticamente

1. Si se disponen de 256 MB de memoria real y el tamaño de página es de 512 KB, el número de marcos se calcula dividiendo el tamaño de la memoria real entre el tamaño de la página:

**Texto, Carta

Descripción generada automáticamente**

1. Si se utilizan 2 KB para cada entrada en la tabla de páginas de un proceso, el tamaño máximo de la tabla de páginas se calcula multiplicando el número máximo de páginas por el tamaño de cada entrada:

Gráfico, Texto

Descripción generada automáticamente con confianza media

**16.-** Como se vio en el ejercicio anterior, la tabla de páginas de un proceso puede alcanzar un tamaño considerablemente grande, que incluso, no podría almacenarse de manera completa en la memoria real. Es por esto que el SO también realiza paginación sobre las tablas de paginas.

Existen varios enfoques para administrar las tablas de páginas:

* Tablas de páginas de 1 nivel.
* Tablas de páginas de 2 niveles.
* Tablas de páginas invertidas.

Explique brevemente como trabajan estos enfoques e indique como se realiza la transformación de la dirección virtual en dirección física.

**Enfoques para Administrar Tablas de Páginas:**

1. **Tablas de Páginas de 1 Nivel:**
   * **Funcionamiento:** Utiliza una única tabla de páginas para mapear direcciones virtuales a físicas.
   * **Transformación de Dirección:** Descompone la dirección virtual en índice de tabla y desplazamiento. El índice se usa para acceder a la tabla, obteniendo el marco físico.
2. **Tablas de Páginas de 2 Niveles:**
   * **Funcionamiento:** Divide la tabla de páginas en dos niveles para estructurar la información.
   * **Transformación de Dirección:** Descompone la dirección virtual en índices de tablas de dos niveles y desplazamiento. Los índices acceden a las tablas para obtener el marco físico.
3. **Tablas de Páginas Invertidas:**
   * **Funcionamiento:** Cada entrada en la tabla de páginas invertida contiene información sobre marcos en memoria física y páginas en espacio virtual.
   * **Transformación de Dirección:** Descompone la dirección virtual en número de página y desplazamiento. El número de página se usa para buscar la entrada, obteniendo el marco físico.

**Transformación de Dirección:** En todos los enfoques, la transformación implica acceder a las tablas de páginas usando índices y números de página. Estas tablas proporcionan información para mapear la dirección virtual a la física, incluyendo el marco. Se agrega el desplazamiento para obtener la dirección física completa.

¿?????

**17.-** Suponga que la tabla de páginas para un proceso que se está ejecutando es la que se muestra a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Página** | **Bit V** | **Bit R** | **Bit M** | **Marco** |
| 0 | 1 | 1 | 0 | 4 |
| 1 | 1 | 1 | 1 | 7 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 2 |
| 4 | 0 | 0 | 0 | - |
| 5 | 1 | 0 | 1 | 0 |

Asumiendo que:

* El tamaño de la pagina es de 512 bytes
* Cada dirección de memoria referencia 1 byte
* Los marcos se encuentras contiguos y en orden en memoria (0, 1, 2.. ) a partir de la dirección real 0.

¿Qué dirección física, si existe, correspondería a cada una de las siguientes direcciones virtuales? (No gestione ningún fallo de página, si se produce)

Se calcula el pasaje a M Fiscia mediante los métodos vistos solo si bit de Valido es true

a) 1052 = Pag 2 Invalido (No existe pagina)

b) 2221 = Pag 4 Invalido (No existe pagina)

c) 5499 = Pag 10 (Excedido)

**d)** 3101= Pag 6 (Excediido)

**18.-** Tamaño de la Página:

La selección del tamaño de la página influye de manera directa sobre el funcionamiento de la memoria virtual. Compare las siguientes situaciones con respecto al tamaño de página, indicando ventajas y desventajas:

* Un tamaño de página pequeño.
* Un tamaño de página grande.
* **Tamaño de Página Pequeño:**
  + **Ventajas:**
    - Menos fragmentación interna.
    - Mayor eficiencia en la utilización de memoria.
    - Mejor ajuste para pequeñas cantidades de datos.
  + **Desventajas:**
    - Mayor sobrecarga en las tablas de páginas.
    - Más fallos de página.
* **Tamaño de Página Grande:**
  + **Ventajas:**
    - Menos sobrecarga en las tablas de páginas.
    - Menos fallos de página.
  + **Desventajas:**
    - Mayor fragmentación interna.
    - Menor eficiencia en la utilización de memoria.
    - Peor ajuste para pequeñas cantidades de datos

*2017*

**19.-** Asignación de marcos a un proceso (Conjunto de trabajo o Working Set):

Con la memoria virtual paginada, no se requiere que todas las páginas de un proceso se encuentren en memoria. El SO debe controlar cuantas páginas de un proceso puede tener en la memoria principal. Existen 2 políticas que se pueden utilizar:

* + Asignación Fija
  + Asignación Dinámica.

1. Describa como trabajan estas 2 políticas.
2. Dada la siguiente tabla de procesos y las paginas que ellos ocupan, y teniéndose 40 marcos en la memoria principal, cuantos marcos le corresponderían a cada proceso si se usa la técnica de Asignación Fija:
   1. Reparto Equitativo
   2. Reparto Proporcional

|  |  |
| --- | --- |
| **Proceso** | **Total de Paginas Usadas** |
| 1 | 15 |
| 2 | 20 |
| 3 | 20 |
| 4 | 8 |

1. ¿Cual de los 2 repartos usados en b) resulto mas eficiente? ¿Por qué?

a)

1. **Asignación Fija:**
   * **Trabajo Inicial:** Se asigna a cada proceso un número fijo de marcos en la memoria principal independientemente de su necesidad real.
   * **Ventajas:** Simplicidad en la implementación.
   * **Desventajas:** Puede conducir a una asignación ineficiente de recursos, ya que algunos procesos pueden necesitar más marcos que otros.
2. **Asignación Dinámica:**
   * **Trabajo Dinámico:** Los marcos se asignan dinámicamente a medida que el proceso los necesita.
   * **Ventajas:** Mayor flexibilidad, ya que los marcos se asignan según las necesidades reales del proceso.
   * **Desventajas:** Mayor complejidad en la implementación y mayor sobrecarga.

**b) Asignación Fija con 40 Marcos:**

i) **Reparto Equitativo:** Todos los procesos obtienen una cantidad igual de marcos: 40/4​=10 marcos por proceso.

ii) **Reparto Proporcional:**Se asignan marcos en proporción al número total de páginas usadas por cada proceso:

* + Proceso 1: 1563×40≈9.52≈106315​×40≈9.52≈9marcos.
  + Proceso 2: 2063×40≈12.7≈136320​×40≈12.7≈13 marcos.
  + Proceso 3: 2063×40≈12.7≈136320​×40≈12.7≈13 marcos.
  + Proceso 4: 863×40≈5.08≈5638​×40≈5.08≈5 marcos.

**c) Eficiencia del Reparto:**

* La eficiencia dependerá de cómo se utilicen los recursos. Si algún proceso no utiliza todos los marcos asignados, podría haber desperdicio. En este caso, se puede evaluar la eficiencia comparando el número real de marcos utilizados por cada proceso con los asignados.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**Formulas de b) Reparto variable**

**20.-** Reemplazo de páginas (selección de una victima):

¿Qué sucede cuando todos los marcos en la memoria principal están usados por las páginas de los procesos y se produce en fallo de página? El SO debe seleccionar una de las páginas que se encuentra en memoria como victima, y ser reemplazada por la nueva página que produjo el fallo.

Considere los siguientes algoritmos de selección de victimas básicos:

* LRU
* FIFO
* OPT (Optimo)
* Segunda Chance

1. Clasifique estos algoritmos de malo a bueno de acuerdo a la tasa de fallos de página que se obtienen al utilizarlos.
2. Analice su funcionamiento. ¿Como los implementaría?
3. Sabemos que la pagina a ser reemplaza puede estar modificada. ¿Qué acciones debe llevar el SO cuando se encuentra ante esta situación?

a) .Clasificación de los algoritmos de selección de víctimas de malo a bueno según la tasa de fallos de página:

1. **FIFO (First-In-First-Out):** Malo - FIFO tiende a tener una tasa de fallos más alta en comparación con otros algoritmos más avanzados, ya que no tiene en cuenta el uso reciente de las páginas.
2. **Segunda Oportunidad (Segunda Chance):** Moderado - Mejora respecto a FIFO al considerar si una página ha sido referenciada recientemente, pero aún puede no ser tan efectivo como otros algoritmos más avanzados.
3. **LRU (Least Recently Used):** Bueno - LRU tiende a tener una tasa de fallos más baja en comparación con FIFO y Second Chance, ya que tiene en cuenta el historial de uso reciente de las páginas.
4. **OPT (Óptimo):** El mejor - OPT siempre seleccionará la página que no se usará por más tiempo en el futuro, lo que teóricamente proporciona la tasa de fallos más baja. Sin embargo, en la práctica, OPT es difícil de implementar completamente porque requiere conocimiento del futuro uso de las páginas.

b) **Análisis de funcionamiento:**

1. **FIFO (First-In-First-Out):** Reemplaza la página más antigua en la cola. Es fácil de implementar utilizando una estructura de cola simple. Sin embargo, puede tener un rendimiento deficiente en situaciones en las que el uso de páginas no sigue el patrón FIFO.
2. **Segunda Oportunidad (Segunda Chance):** Similar a FIFO, pero utiliza un bit de referencia para distinguir entre páginas referenciadas y no referenciadas. Si una página tiene un bit de referencia, se le da "una segunda oportunidad" y se mueve al final de la cola. Esto se puede implementar utilizando una cola circular y un bit de referencia.
3. **LRU (Least Recently Used):** Mantiene un registro del orden en que se usaron las páginas y reemplaza la que no se ha usado durante más tiempo. Puede ser implementado utilizando una estructura de datos más compleja, como una lista enlazada o un contador de tiempo, lo que puede tener un costo adicional.
4. **OPT (Óptimo):** Selecciona la página que no se usará por más tiempo en el futuro. En la práctica, es difícil de implementar ya que requiere conocimiento del futuro comportamiento del programa, lo que no es realista en un entorno de sistema operativo.

c) **Página modificada:** Cuando una página a ser reemplazada está modificada, el sistema operativo debe asegurarse de que los cambios se reflejen en el almacenamiento persistente (generalmente en disco). Esto implica escribir la página modificada de vuelta al almacenamiento secundario antes de reemplazarla en la memoria principal. Este proceso se conoce como escritura de vuelta o escritura de actualización. Sin hacer esto, se perderían los cambios realizados en la página y se produciría una inconsistencia en los datos.

**21.-** Alcance del reemplazo

Al momento de tener que seleccionar una pagina victima, el SO puede optar por 2 políticas a utilizar:

* + Reemplazo local
  + Reemplazo global

1. Describa como trabajan estas 2 políticas.
2. ¿Es posible utilizar la política de “Asignación Fija” de marcos junto con la política de “Reemplazo Global? Justifique.

a)

1. **Reemplazo Local:**
   * En el reemplazo local, también conocido como reemplazo por proceso, cada proceso tiene su propia tabla de páginas.
   * Cuando se necesita seleccionar una página víctima para reemplazo, se realiza la selección dentro de las páginas asignadas a ese proceso específico.
   * Este enfoque permite que diferentes procesos tengan sus propias políticas de reemplazo, adaptadas a sus patrones de acceso a memoria.
2. **Reemplazo Global:**
   * En el reemplazo global, también conocido como reemplazo por sistema, todas las páginas en memoria forman un único conjunto global.
   * Cuando se debe seleccionar una página víctima, la selección se realiza desde todas las páginas en memoria, independientemente de a qué proceso pertenecen.
   * Este enfoque permite una asignación más dinámica de marcos entre procesos, ya que cualquier marco puede ser seleccionado para reemplazo sin restricciones locales.

b) **Posibilidad de utilizar "Asignación Fija" con "Reemplazo Global":**

En general, es posible utilizar la política de "Asignación Fija" de marcos junto con la política de "Reemplazo Global". La asignación fija se refiere a asignar un conjunto específico de marcos a cada proceso. Esto implica que cada proceso tiene su propio conjunto de marcos en la memoria principal.

En el contexto de reemplazo global, aunque los marcos están asignados fijamente a cada proceso, la elección de qué página reemplazar se realiza globalmente, considerando todas las páginas en memoria sin restricciones locales.

**22.-** Considere la siguiente secuencia de referencias de páginas:

1, 2, 15, 4, 6, 2, 1, 5, 6, 10, 4, 6, 7, 9, 1, 6, 12, 11, 12, 2, 3, 1, 8, 1, 13, 14, 15, 3, 8

a

1. Si se disponen de 5 marcos. ¿Cuántos fallos de página se producirán si se utilizan las siguientes técnicas de selección de victima? (Considere una política de Asignación Dinámica y Reemplazo Global)
   1. Segunda Chance
   2. FIFO
   3. LRU
   4. OPT
2. Suponiendo que cada atención de un fallo se pagina requiere de 0,1 seg. Calcular el tiempo consumido por atención a los fallos de páginas para los algoritmos de a). Es 0.1\* cant fallos, no se calcula

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Marco/Pagina | 1 | 2 | 15 | 4 | 6 | 2 | 1 | 5 | 6 | 10 | 4 | 6 | 7 | 9 | 1 | 6 | 12 | 11 | 12 | 2 | 3 | 1 | 8 | 1 | 13 | 14 | 15 | 3 | 8 |
| Marco 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 3 |
| Marco 2 |  | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 12 | 12 | 12 | 12 | 12 | 12 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Marco 3 |  |  | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 7 | 7 | 7 | 7 | 7 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 13 | 13 | 13 | 13 | 13 |
| Marco 4 |  |  |  | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 9 | 9 | 9 | 9 | 9 | 9 | 2 | 2 | 2 | 2 | 2 | 2 | 14 | 14 | 14 | 14 |
| Marco 5 |  |  |  |  | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 15 | 15 | 15 |
| Fallo | X | X | X | X | X |  |  | X |  | X |  |  | X | X | X | X | X | X |  | X | X | X | X |  | X | X | X | X |  |
| FIFO |  |  | Fallos : 21 | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Marco/Pagina | 1 | 2 | 15 | 4 | 6 | 2 | 1 | 5 | 6 | 10 | 4 | 6 | 7 | 9 | 1 | 6 | 12 | 11 | 12 | 2 | 3 | 1 | 8 | 1 | 13 | 14 | 15 | 3 | 8 |
| Marco 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1\* | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 12 | 12 | 12\* | 12 | 12 | 12 | 12 | 12 | 13 | 13 | 13 | 13 | 13 |
| Marco 2 |  | 2 | 2 | 2 | 2 | 2\* | 2\* | 2 | 2 | 2 | 2 | 2 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 15 | 15 | 15 |
| Marco 3 |  |  | 15 | 15 | 15 | 15 | 15 | 5 | 5 | 5 | 5 | 5 | 5 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3\* | 3\* |
| Marco 4 |  |  |  | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4\* | 4\* | 4 | 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1\* | 1 | 1\* | 1 | 14 | 14 | 14 | 14 |
| Marco 5 |  |  |  |  | 6 | 6 | 6 | 6 | 6\* | 6 | 6 | 6\* | 6 | 6 | 6 | 6\* | 6 | 11 | 11 | 11 | 11 | 11 | 8 | 8 | 8 | 8 | 8 | 8 | 8\* |
| Fallo | X | X | X | X | X |  |  | x |  | x |  |  | x | x | x |  | X | x |  | x | x |  | x |  | x | x | X |  |  |
| Segunda Chance | |  | Fallos : 18 | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Marco/Pagina | 1 | 2 | 15 | 4 | 6 | 2 | 1 | 5 | 6 | 10 | 4 | 6 | 7 | 9 | 1 | 6 | 12 | 11 | 12 | 2 | 3 | 1 | 8 | 1 | 13 | 14 | 15 | 3 | 8 |
| Marco 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 13 | 13 | 13 | 13 | 13 |
| Marco 2 |  | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| Marco 3 |  |  | 15 | 15 | 15 | 15 | 15 | 5 | 5 | 10 | 10 | 10 | 7 | 9 | 9 | 9 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 14 | 15 | 15 | 15 |
| Marco 4 |  |  |  | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
| Marco 5 |  |  |  |  | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Fallo | x | x | x | x | x |  |  | x |  | x |  |  | x | x |  |  | x | x |  |  | x |  | x |  | x | x | x |  |  |
| OPT |  |  | Fallos: | | 16 |  |  |  |  |  |  |  | Reemplazo terminados random | | | | | | | | | |  |  |  |  |  |  |  |

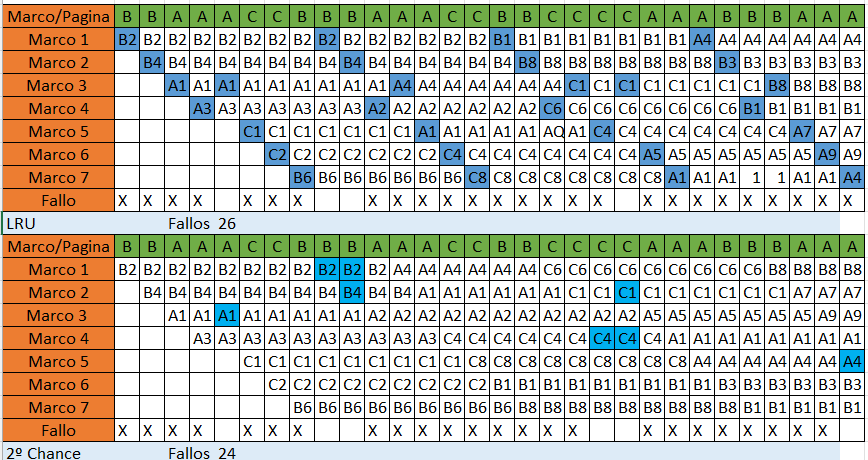
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Marco/Pagina | 1 | 2 | 15 | 4 | 6 | 2 | 1 | 5 | 6 | 10 | 4 | 6 | 7 | 9 | 1 | 6 | 12 | 11 | 12 | 2 | 3 | 1 | 8 | 1 | 13 | 14 | 15 | 3 | 8 |
| Marco 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7 | 7 | 7 | 7 | 7 | 11 | 11 | 11 | 11 | 11 | 8 | 8 | 8 | 8 | 8 | 3 | 3 |
| Marco 2 |  | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 4 | 4 | 4 | 4 | 4 | 4 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 13 | 13 | 13 | 13 | 13 |
| Marco 3 |  |  | 15 | 15 | 15 | 15 | 15 | 5 | 5 | 5 | 5 | 5 | 5 | 9 | 9 | 9 | 9 | 9 | 9 | 2 | 2 | 2 | 2 | 2 | 2 | 14 | 14 | 14 | 14 |
| Marco 4 |  |  |  | 4 | 4 | 4 | 4 | 4 | 4 | 10 | 10 | 10 | 10 | 10 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 15 | 15 | 15 |
| Marco 5 |  |  |  |  | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 8 |
| Fallo | x | x | x | x | x |  |  | x |  | x | x |  | x | x | x |  | x | x |  |  | x | x | x |  | x | x | x | x | x |
| LRU |  |  | Fallos : | | 21 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**23.-** Sean los procesos A, B y C tales que necesitan para su ejecución las siguientes páginas:

* **A**: 1, 3, 1, 2, 4, 1, 5, 1, 4, 7, 9, 4
* **B**: 2, 4, 6, 2, 4, 1, 8, 3, 1, 8
* **C**: 1, 2, 4, 8, 6, 1, 4, 1

Si la secuencia de ejecución es tal que los procesos se ejecutan en la siguiente secuencia:

1. **B** demanda 2 páginas
2. **A** demanda 3 páginas
3. **C** demanda 2 páginas
4. **B** demanda 3 páginas
5. **A** demanda 3 páginas
6. **C** demanda 2 páginas
7. **B** demanda 2 páginas
8. **C** demanda 4 páginas
9. **A** demanda 3 páginas
10. **B** demanda 3 páginas
11. **C** termina
12. **A** demanda 3 páginas
13. **B** termina
14. **A** termina
15. Considerando una política de Asignación Dinámica y Reemplazo Global y disponiéndose de 7 marcos. ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de victimas:
    1. LRU ii) Segunda Chance
16. Considerando una política de Asignación Fija con reparto equitativo y Reemplazo Local y disponiéndose de 9 marcos. ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de victimas:
    1. LRU **ii)** Segunda Chance

a)

**Un conjunto de letras blancas en un fondo blanco

Descripción generada automáticamente con confianza media**

**Un conjunto de letras blancas en un fondo blanco

Descripción generada automáticamente con confianza media**

**b)**

**Imagen que contiene Tabla

Descripción generada automáticamente**

**24.-** Sean los procesos A, B y C tales que necesitan para su ejecución las siguientes páginas:

* **A**: 1, 2, 1, 7, 2, 7, 3, 2
* **B**: 1, 2, 5, 2, 1, 4, 5
* **C**: 1, 3, 5, 1, 4, 2, 3

Si la secuencia de ejecución es tal que los procesos se ejecutan en la siguiente manera:

1. **C** demanda 1 página
2. **A** demanda 2 páginas
3. **C** demanda 1 página
4. **B** demanda 1 página
5. **A** demanda 1 página
6. **C** modifica la página 1
7. **B** demanda 2 páginas
8. **A** demanda 1 página
9. **C** demanda 1 página
10. **B** modifica la página 2
11. **A** modifica la página 2
12. **B** demanda 2 páginas
13. **A** demanda 1 página
14. **B** demanda 2 páginas
15. **C** demanda 2 páginas
16. **C** demanda 1 página
17. **A** demanda 1 página
18. **B** termina
19. **A** demanda 2 páginas
20. **C** demanda 1 página
21. **A** termina
22. **C** termina

Considerando una política de Asignación Dinámica y Reemplazo Global y disponiéndose de 7 marcos, debiéndose guardar 1 marco para la gestión de descarga asincrónica de paginas modificadas ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de victima:

1. Segunda Chance
2. FIFO
3. LRU

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Nota = Al sacar un modificado se requieren 2 ciclos, que SE ANOTA CON XD (X= FALLO, D= DELAY) | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | 1 | 2 | 2 | 3 | 4 | 5 | 6 | 7 | 7 | 8 | 9 | 10 | 11 | 12 | 12 | 13 | 14 | 14 | 15 | 15 | 16 | 17 | 19 | 19 | 20 |
| Marco/Pagina | C | A | A | C | B | A | C | B | B | A | C | B | A | B | B | A | B | B | C | C | C | A | A | A | C |
| Marco 1 | C1 | C1 | C1 | C1 | C1 | C1 | C1- | C1- |  |  |  |  |  |  |  |  |  |  |  | C4 | C4 | C4 | C4 | C4 | C4 |
| Marco 2 |  | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 | B5 | B5 | B5 | B5 | B5 | A3 | A3 | A3 |
| Marco 3 |  |  | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C1 | C1 | C1 | C1 | C1 | C1 | C3 |
| Marco 4 |  |  |  | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | A2- | A2- | A2- | A2- | A2- | A2- | A2- | A2- | A2- | A2- | A2- | A2 | A2- |
| Marco 5 |  |  |  |  | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | A7 | A7 | A7 | A7 |
| Marco 6 |  |  |  |  |  |  |  | B2 | B2 | B2 | B2 | B2- |  | B2- | B2- | B2- | B2- | B2- | B2- | B2- |  |  |  |  |  |
| Marco 7 |  |  |  |  |  |  |  |  | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B4 | B4 | B4 |  | C2 | C2 | C2 | C2 | C2 |
| Fallo | X | X | X | X | X |  |  | X | X | X | X | D | XD |  |  |  | X | X | X | XD |  | X | X |  |  |
| 2º Chance |  |  | Fallos : | | 16 |  |  |  |  | Pagina Modificada | | | | | | - |  |  |  | Espacio Libre | | | | |  |
|  | Revisar (La idea gral esta, error en algun lado) | | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 2 | 3 | | 4 | | 5 | | 6 | 7 | 7 | 8 | 9 | 10 | | 11 | | 12 | | 12 | | 13 | | 14 | 14 | 15 | 15 | 16 | 17 | 19 | 19 | 20 |
| Marco/Pagina | C | A | A | C | | B | | A | | C | B | B | A | C | B | | A | | B | | B | | A | | B | B | C | C | C | A | A | A | C |
| Marco 1 | C1 | C1 | C1 | C1 | | C1 | | C1 | | C1- | C1- | C1 | C1 | C1 | C1 | | C1- | | C1 | |  | |  | |  | B5 | B5 | B5 | B5 | B5 | B5 | A2 | A2 |
| Marco 2 |  | A1 | A1 | A1 | | A1 | | A1 | | A1 | A1 | A1 | A1 | A1 | A1 | | A2- | | A2- | | A2- | | A2- | | A2- |  |  | C4 | C4 | C4 | C4 | C4 | C4 |
| Marco 3 |  |  | A2 | A2 | | A2 | | A2 | | A2 | A2 | B5 | B5 | B5 | B5 | | B5 | | B5 | | B5 | | B5 | | B4 | B4 | B4 | B4 | B4 | B4 | A3 | A3 | A3 |
| Marco 4 |  |  |  | C3 | | C3 | | C3 | | C3 | C3 | C3 | A7 | A7 | A7 | | A7 | | A7 | | A7 | | A7 | | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 |
| Marco 5 |  |  |  |  | | B1 | | B1 | | B1 | B1 | B1 | B1 | C5 | C5 | | C5 | | C5 | | C5 | | C5 | | C5 | C5 | C1 | C1 | C1 | C1 | C1 | C1 | C3 |
| Marco 6 |  |  |  |  | |  | |  | |  | B2 | B2 | B2 | B2 | B2- | | B2 | | B2- | | B2 | | B2 | | B2 | B2 | B2- |  |  |  |  |  |  |
| Marco 7 |  |  |  |  | |  | |  | |  |  |  |  |  |  | |  | |  | | B1 | | B1 | | B1 | B1 | B1 | B1 | C2 | C2 | C2 | C2 | C2 |
| Fallo | X | X | X | X | | X | |  | |  | X | X | X | X |  | | X | |  | | XD | |  | | X | XD | X | XD | X |  | X | X | X |
| LRU |  | F | 19 | |  | |  | | Pagina Modificada | | | | | | | - | |  | |  | |  | | Espacio Libre | | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 2 | 3 | 4 | 5 | 6 | 7 | 7 | 8 | 9 | 10 | 11 | 12 | 12 | 13 | 14 | 14 | 15 | 15 | 16 | 17 | 19 | 19 | 20 |
| Marco/Pagina | C | A | A | C | B | A | C | B | B | A | C | B | A | B | B | A | B | B | C | C | C | A | A | A | C |
| Marco 1 | C1 | C1 | C1 | C1 | C1 | C1 | C1- | C1- |  |  |  |  |  |  |  | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 | A7 |
| Marco 2 |  | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 | C5 |
| Marco 3 |  |  | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2- | A2 | A2- |  |  |  |  |  | C2 | C2 | C2 | C2 | C2 |
| Marco 4 |  |  |  | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C3 | C1 | C1 | C1 | C1 | C1 | C1 | C3 |
| Marco 5 |  |  |  |  | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | C4 |  |  |  |  |  |
| Marco 6 |  |  |  |  |  |  |  | B2 | B2 | B2 | B2 | B2- | B2- | B2- | B2- | B2- | B2- | B2- | B2- | B2- | B2- | B2F | A3 | A3A3 |  |
| Marco 7 |  |  |  |  |  |  |  |  | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 | B5 |  |  | A2 | A2 |
| Fallo | X | X | X | X | X |  |  | X | XD |  | X |  |  |  |  | XD |  |  | X | X | XD |  | X | X | X |
| FIFO |  |  | Fallos : | | 14 |  |  |  |  | Pagina Modificada | | | | | | - |  |  |  | Espacio Libre | | | | |  |

**25.-** Hiperpaginación (Trashing)

1. ¿Qué es?
2. ¿Cuáles pueden ser los motivos que la causan?
3. ¿Cómo la detecta el SO?
4. Una vez que lo detecta, ¿qué acciones puede tomar el SO para eliminar este problema?

**a) La** hiperpaginación, también conocida como "trashing", es una situación en la que el sistema operativo se encuentra tan ocupado intercambiando páginas entre la memoria principal y el almacenamiento secundario (por ejemplo, el disco) que queda poco tiempo para la ejecución real de programas. En otras palabras, el sistema está ocupado moviendo constantemente páginas hacia adelante y hacia atrás entre la memoria principal y el almacenamiento secundario, lo que lleva a una baja eficiencia y rendimiento.

**b)** La hiperpaginación puede ser causada por varios motivos, entre ellos:

1. **Insuficiente cantidad de memoria principal:** Cuando la memoria principal no es suficiente para satisfacer las demandas de múltiples procesos en ejecución, se produce una carga excesiva en el sistema de paginación.
2. **Demanda excesiva de páginas:** Si los programas generan una demanda constante de páginas y la memoria principal es limitada, se produce un ciclo continuo de carga y descarga de páginas.

**c) ¿Cómo la detecta el SO?** El sistema operativo puede detectar la hiperpaginación mediante el monitoreo de la tasa de fallos de página. Si la tasa de fallos de página es extremadamente alta y el sistema pasa la mayor parte del tiempo moviendo páginas entre la memoria y el almacenamiento secundario en lugar de ejecutar instrucciones de programas, es probable que haya trashing.

**d) Una vez que lo detecta, ¿qué acciones puede tomar el SO para eliminar este problema?** Para abordar la hiperpaginación, el sistema operativo puede tomar las siguientes acciones:

1. **Ajustar la prioridad de los procesos:** Redistribuir la prioridad entre los procesos para dar más tiempo de CPU a aquellos que realmente lo necesitan.
2. **Ajustar el tamaño de los marcos de página:** Aumentar el tamaño de los marcos de página puede reducir la tasa de fallos de página, ya que menos páginas se necesitarían para almacenar la misma cantidad de datos.
3. **Suspender procesos:** Puede suspender temporalmente algunos procesos para liberar recursos y reducir la demanda de páginas.
4. **Optimizar algoritmos de reemplazo de página:** Utilizar algoritmos de reemplazo de página más eficientes o ajustar parámetros de configuración puede ayudar a mejorar la gestión de la memoria y reducir la hiperpaginación.

**26.-** Considere un sistema cuya memoria principal se administra mediante la técnica de paginación por demanda que utiliza un dispositivo de paginación, algoritmo de reemplazo global LRU y una política de asignación que reparte marcos equitativamente entre los procesos. El nivel de multiprogramación es actualmente, de 4.

Ante las siguientes mediciones:

1. Uso de CPU del 13%, uso del dispositivo de paginación del 97%.
2. Uso de CPU del 87%, uso del dispositivo de paginación del 3%.
3. Uso de CPU del 13%, uso del dispositivo de paginación del 3%.

Analizar:

* + ¿Qué sucede en cada caso?
  + ¿Puede incrementarse el nivel de multiprogramación para aumentar el uso de la CPU?
  + ¿La paginación está siendo útil para mejorar el rendimiento del sistema?

a) **Uso de CPU del 13%, uso del dispositivo de paginación del 97%:**

* Bajo uso de CPU indica falta de atención a procesos.
* Alto uso del dispositivo de paginación sugiere carga en la memoria y fallos de página.

b) **Uso de CPU del 87%, uso del dispositivo de paginación del 3%:**

* Alto uso de CPU indica rendimiento eficiente.
* Bajo uso del dispositivo de paginación sugiere pocos fallos de página y uso efectivo de la memoria.

c) **Uso de CPU del 13%, uso del dispositivo de paginación del 3%:**

* Bajo uso tanto de CPU como de dispositivo de paginación indica falta de actividad significativa.

**Análisis General:**

* Incrementar el nivel de multiprogramación puede aumentar el uso de la CPU, pero también puede generar más competencia por recursos y aumentar el uso del dispositivo de paginación.
* La paginación es útil en situaciones de baja eficiencia en el uso de la CPU y alto uso del dispositivo de paginación, pero también contribuye a un rendimiento eficiente al minimizar los fallos de página en situaciones de alto rendimiento de la CPU y bajo uso del dispositivo de paginación.

**27.-** Considere un sistema cuya memoria principal se administra mediante la técnica de paginación por demanda. Considere las siguientes medidas de utilización:

* + Utilización del procesador: 20%
  + Utilización del dispositivo de paginación: 97,7%
  + Utilización de otros dispositivos de E/S: 5%

Cuales de las siguientes acciones pueden mejorar la utilización del procesador:

1. Instalar un procesador mas rápido
2. Instalar un dispositivo de paginación mayor
3. Incrementar el grado de multiprogramación
4. Instalar mas memoria principal
5. Decrementar el quantum para cada proceso

a) Instalar un procesador más rápido: No sirve en este caso.

b) Instalar un dispositivo de paginación mayor: Esto podría ayudar a reducir la velocidad de ocurrencia del trashing, pero los fallos los seguis teniendo

c) Incrementar el grado de multiprogramación: Esto implica tener más procesos en memoria al mismo tiempo. No es lo recomendable en este caso

d) Instalar más memoria principal: Aumentar la memoria principal puede reducir la necesidad de paginación, lo que puede mejorar la utilización del procesador al reducir las transferencias de página.

e) Decrementar el quantum para cada proceso: Reducir el quantum empeora las cosas

**28.-** La siguiente formula describe el tiempo de acceso efectivo a la memoria al utilizar paginación para la implementación de la memoria virtual:

Donde:

*TAE*  *At*  (1  *p*) \* *Am*  *p* \* (*Tf + Am* )

TAE = tiempo de acceso efectivo

p = taza de fallo de pagina (0 <= p <=1

Am = tiempo de acceso a la memoria real

Tf = tiempo se atención de una fallo de pagina

At = tiempo de acceso a la tabla de paginas. Es igual al tiempo de acceso a la memoria (Am) si la entrada de la tabla de páginas no se encuentra en la TLB.

Suponga que tenemos una memoria virtual paginada, con tabla de paginas de 1 nivel, y donde la tabla de páginas se encuentra completamente en la memoria.

Servir una falla de página tarda 300 nanosegundos si hay disponible un marco vacío o si la página reemplazada no se ha modificado, y 500 nanosegundos si se ha modificado. El tiempo de acceso a memoria es de 20 nanosegundos y el de acceso a la TLB es de 1 nanosegundo

1. Si suponemos una taza de fallos de página de 0,3 y que siempre contamos con un marco libre para atender el fallo ¿Cual será el TAE si el 50% de las veces la entrada de la tabla de páginas se encuentra en la TLB (hit)?
2. Si suponemos una taza de fallos de página de 0,3; que el 70% de las ocasiones la pagina a reemplazar se encuentra modificada. ¿Cual será el TAE si el 60% de las veces la entrada de la tabla de páginas se encuentra en la TLB (hit)?
3. Si suponemos que el 60% de las veces la pagina a reemplazar esta modificada, el 100% de las veces la entrada de la tabla de páginas requerida se encuentra en la TLB (hit) y se espera un TAE menor a 200 nanosegundos.

¿Cuál es la máxima tasa aceptable de fallas de página?

Formula

*TAE*  *At*  (1  *p*) \* *Am*  *p* \* (*Tf + Am* )

1. At = 0.5\*1ns + 0.5\*20ns

Am= 20ns

P=0.3

Tf=300ns

*TAE*=(0.5⋅1+0.5⋅*Am*)+(1−0.3)⋅*Am*+0.3⋅(*Tf*+*Am*) = 120.5 ns = 121 ns

1. At = 0.6\*1ns + 0.4\*20ns

Am= 20ns

P=0.3

Tf=300ns\*0,3 + 0,7\*500ns

*TAE=(0.6⋅1+0.4⋅Am)+(1−0.3)⋅Am+0.3⋅(Tf+Am) =154,6 ns = 161 ns*

1. At = 1ns

Am= 20ns

P=0.3

Tf=300ns\*0,4 + 500\*0,6ns

*200*=(1ns)+(1−p)⋅*20*+p⋅(*440*)

**P=0,41**

**29.-** Anomalía de Belady

1. ¿Qué es?
2. Dada la siguiente secuencia de referencias a paginas: 3, 2, 1, 0, 3, 2, 4, 3, 2, 1, 0, 4
   1. Calcule la cantidad de fallos de páginas si se cuentan con 3 marcos y se utiliza el algoritmo de reemplazo FIFO
   2. Calcule la cantidad de fallos de páginas si se cuentan con 4 marcos y se utiliza el algoritmo de reemplazo FIFO

Analice la situación

*- Cursada 2017*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

a)La anomalía de Belady se refiere a una situación en la que aumentar la cantidad de marcos de página en un sistema de memoria paginada no necesariamente disminuye la tasa de fallos de página. En otras palabras, puede haber casos en los que, al agregar más memoria, la tasa de fallos de página aumenta en lugar de disminuir. Esta es una observación no intuitiva que se da en algunos algoritmos de reemplazo de página.

b)

Imagen que contiene Texto

Descripción generada automáticamenteImagen que contiene Tabla

Descripción generada automáticamenteEn este caso específico, la secuencia de referencias y el algoritmo FIFO resultaron en la misma cantidad de fallos de página con 3 y 4 marcos. Esto no siempre ocurre, y la anomalía de Belady se manifiesta cuando, al aumentar la cantidad de marcos, la tasa de fallos de página no disminuye o incluso aumenta. La anomalía destaca que algunos algoritmos de reemplazo de página pueden comportarse de manera no intuitiva en relación con la cantidad de memoria disponible.

**30.-** Considere el siguiente programa:

#define Size 64

int A[Size; Size], B[Size; Size], C[Size; Size]; int register i, j;

for (j = 0; j < Size; j ++)

for (i = 0; i < Size; i++)

C[i; j] = A[i; j] + B[i; j];

Si asumimos que el programa se ejecuta en un sistema que utiliza paginación por demanda para administrar la memoria, donde cada pagina es de 1Kb. Cada número entero (int) ocupa 4 bytes. Es claro que cada matriz requiere de 16 páginas para almacenarse. Por ejemplo: A[0,0]..A[0,63], A[1,0]..A[1,63], A[2,0]..A[2,63] y

A[3,0]..A[3,63] se almacenara en la primer pagina.

Asumamos que el sistema utiliza un working set de 4 marcos para este proceso. Uno de los 4 marcos es utilizado por el programa y los otros 3 se utilizan para datos (las matrices). También asumamos que para los índices “i” y “j” se utilizan 2 registros, por lo que no es necesario el acceso a la memoria para estas 2 variables.

1. Analizar cuantos fallos de paginas ocurren al ejecutar el programa (considere las veces que se ejecuta C[i,j] = A[i,j] + B[i,j])
2. Puede ser modificado el programa para minimizar el número de fallos de páginas. En caso de ser posible indicar la cantidad de fallos de fallos de páginas que ocurre

Texto

Descripción generada automáticamente

b) Para minimizar el número de fallos de páginas en este programa, se puede aplicar la técnica de bloqueo de bloques (blocking) para mejorar la localidad espacial y temporal. La idea es dividir las matrices en bloques más pequeños y procesar esos bloques de manera iterativa. Esto ayuda a reducir la cantidad de datos que se acceden en cada iteración y, por lo tanto, puede reducir la cantidad de fallos de página.

#define Size 64

#define BlockSize 8 // Tamaño del bloque

int A[Size][Size], B[Size][Size], C[Size][Size];

int register i, j, ii, jj, iii, jjj;

for (jj = 0; jj < Size; jj += BlockSize) {

for (ii = 0; ii < Size; ii += BlockSize) {

for (j = jj; j < jj + BlockSize; j++) {

for (i = ii; i < ii + BlockSize; i++) {

C[i][j] = A[i][j] + B[i][j];

}

}

}

}

**31.-** Considere las siguientes secuencias de referencias a páginas de los procesos A y B, donde se muestra en instante de tiempo en el que ocurrió cada una (1 a 78):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Proceso A | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 1 | 2 | 1 | 3 | 4 | 3 | 3 | 2 | 1 | 1 | 3 | 4 | 5 | 6 | 1 | 2 | 1 | 3 | 3 | 4 | 5 | 6 | 6 | 6 | 5 | 4 | 3 | 1 | 1 | 2 | 3 | 4 | 5 | 4 | 3 | 2 | 1 | 1 | 1 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 |
| 2 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 6 | 5 | 3 | 1 | 2 | 3 | 4 | 5 | 6 | 5 | 1 | 1 | 2 | 3 | 4 | 5 | 4 | 2 | 1 | 1 | 2 | 3 | 4 | 5 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Proceso B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 1 | 2 | 3 | 3 | 3 | 4 | 4 | 5 | 1 | 1 | 1 | 2 | 3 | 4 | 4 | 4 | 4 | 1 | 1 | 2 | 3 | 6 | 5 | 6 | 5 | 4 | 6 | 1 | 1 | 1 | 1 | 4 | 5 | 1 | 3 | 2 | 1 | 1 | 2 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 |
| 2 | 2 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 5 | 4 | 6 | 1 | 1 | 1 | 2 | 3 | 3 | 3 | 4 | 5 | 5 | 5 | 1 | 6 | 2 | 3 | 1 | 2 | 3 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 3 |

1. Considerando una ventana ∆=5, indique cual seria el conjunto de trabajo de los procesos A y B en el instante 24 (WSA(24) y WSB(24))
2. Considerando una ventana ∆=5, indique cual seria el conjunto de trabajo de los procesos A y B en el instante 60 (WSA(60) y WSB(60))
3. Para el los WS obtenidos en el inciso a), si contamos con 8 frames en el sistema ¿Se puede indicar que estamos ante una situación de trashing? ¿Y si contáramos con 6 frames?
4. Considerando únicamente el proceso A, y suponiendo que al mismo se le asignaron inicialmente 4 marcos, donde el de reemplazo de paginas es realizado considerando el algoritmo FIFO. ¿Cuál será la taza de fallos en el instante 38 de páginas suponiendo que la misma se calcula contando los fallos de páginas que ocurrieron en las últimas 10 unidades de tiempo?
5. Para el valor obtenido en el inciso d), si suponemos que el S.O. utiliza como limites superior e inferior de taza de fallos de paginas los valores 2 y 5 respectivamente ¿Qué acción podría tomar el S.O. respecto a la cantidad de marcos asignados al proceso?

a,b) Resaltada la tabla

1. Con 8 y 6 marcos no, ya que solo tenemos utilizados 5 procesos en total

Si D ⟩ m habrá thrashing

m = cantidad de frames disponibles

D = demanda total de frames (5)

1. Imagen en blanco y negro de un teclado

   Descripción generada automáticamente con confianza baja.
2. Dado que la tasa de fallos es 38 y los límites superior e inferior son 5 y 2 respectivamente:

La tasa de fallos supera el límite superior (5). En este caso, el S.O. podría interpretar que el proceso está utilizando demasiados recursos o que el tamaño de la memoria asignada es insuficiente para contener el WS. Una posible acción podría ser aumentar la cant. de marcos asignados al proceso para reducir los fallos

Principio del formulario

*- Cursada 2017*

*\_\_\_\_\_\_\_\_\_\_\_\_*

2º Chance, Resolución correcta

Imagen que contiene biombo, pájaro, reloj, cuarto

Descripción generada automáticamenteImagen en blanco y negro

Descripción generada automáticamente con confianza media

<https://www.youtube.com/watch?v=FJMl7v_x6DI&ab_channel=SylvinaEnriquez>

Imagen que contiene Calendario

Descripción generada automáticamenteEnlace