Informe Caso 2

David Alejandro Abril - 202224328 - d.abrilm

Monica González Vargas – 202116872 – m.gonzalezv2

• Descripción del algoritmo usado para generar las referencias de página (modo uno)

En el algoritmo de referencia se tienen como parámetros el tamaño de pagina y el nombre del archivo que guarda la imagen con el mensaje escondido. Se calcula la cantidad total de referencias necesarias para representar tanto la imagen como el mensaje. El cálculo del número de páginas necesarias se realiza basándose en las dimensiones de la imagen y el tamaño del mensaje, para determinar el espacio que ocupa en memoria. Se verifica si el directorio de salida ("salida") existe, y si no, se crea. Luego, se crea y escribe un archivo llamado referencias.txt, en el cual se guardará información de las referencias. Se escriben datos como el tamaño de la página, el número de filas (alto de la imagen), el número de columnas (ancho de la imagen), el número total de referencias, y el total de páginas.

Referencias de la imagen: Se generan referencias a los píxeles de la imagen. Cada píxel está compuesto por 3 canales de color (R, G, B). La referencia incluye la fila y columna del píxel, el canal de color, la página virtual en la que está almacenado y el desplazamiento dentro de la página.

Referencias del mensaje: Después de las primeras 16 referencias que corresponden exclusivamente a la imagen, el algoritmo alterna entre referencias de imagen y de mensaje. Las referencias de mensaje se identifican por el índice del byte del mensaje, la página en la que está almacenado y el desplazamiento dentro de esa página. La alternancia se gestiona mediante un índice separado para el mensaje (indiceMensaje) y el control de los canales de color para la imagen. Los datos se distribuyen en bloques o "páginas". Cada vez que el desplazamiento llega al tamaño de la página, se incrementa el número de página virtual y se reinicia el desplazamiento a 0. Este proceso ocurre tanto para las referencias de imagen como para las de mensaje.

Las referencias de la imagen siguen el formato: Imagen[fila][columna].canal, página, desplazamiento, R.

Las referencias del mensaje siguen el formato: Mensaje[byte], página, desplazamiento, W.

• Descripción de las estructuras de datos usadas para simular el comportamiento del sistema de paginación y cómo usa dichas estructuras (cuándo se actualizan, con base en qué y en qué consiste la actualización).

Estructura de Datos Utilizadas:

LinkedList<Integer> paginasCargadas:

Esta lista enlazada simula los marcos de página en memoria. Cada entrada en la lista representa una página cargada en uno de los marcos.

Uso: La lista es usada para almacenar las páginas que están actualmente cargadas en la memoria. Al procesar un acceso a una página, se verifica si la página ya está en esta lista (acceso exitoso) o si es necesario cargarla (falla de página). Si la lista alcanza su capacidad máxima, se utiliza un algoritmo de reemplazo de página.

Actualización: La lista se actualiza en dos casos:

Falla de página: Si la página no está en la lista y hay espacio disponible, se agrega al final de la lista. Reemplazo de página: Si la lista está llena, se reemplaza una de las páginas en la lista usando una política basada en el bit de referencia (explicado más abajo).

Procesador de accesos: Este hilo lee el archivo de accesos (referencias.txt) línea por línea, simulando accesos a páginas con un pequeño retardo (Thread.sleep(1) para simular 1 milisegundo). Por cada acceso, llama al método procesarAcceso() para verificar si la página está cargada o si se necesita cargar en memoria.

Actualizador de bits: Este hilo resetea periódicamente los bits de referencia (referenciaBit[]), cada 2 milisegundos, para asegurarse de que las páginas referenciadas recientemente no permanezcan indefinidamente en memoria sin ser reemplazadas.

reemplazando la página más antigua. Cada 2 milisegundos, el hilo de actualización resetea todos los bits de referencia (referenciaBit[]), haciendo que las páginas que no sean accedidas en este período sean candidatas para reemplazo en el siguiente ciclo.

Cuando se accede a una página, si ya está cargada en los marcos (paginasCargadas), el acceso es exitoso, y el bit de referencia correspondiente se marca como true, indicando que la página ha sido utilizada.

Si la página no está en memoria, ocurre una falla de página. En este caso: Si hay espacio en los marcos, la página se agrega a la lista de paginasCargadas.Si no hay espacio, se ejecuta el proceso de reemplazo de página.

Proceso de Reemplazo de Página: El reemplazo de páginas sigue un enfoque de segunda oportunidad: Primera búsqueda: Se recorre la lista de páginas cargadas y se revisa el bit de referencia. Si encuentra una página cuyo bit de referencia es false, esa página es reemplazada. Si encuentra una página cuyo bit de referencia es true, le da una "segunda oportunidad", reseteando su bit de referencia a false y dejando esa página en memoria. Luego pasa a revisar la siguiente página. Si ninguna página tiene el bit de referencia en false durante el primer recorrido, se reemplaza la primera página en la lista (FIFO).

Este comportamiento garantiza que las páginas que han sido referenciadas recientemente permanezcan más tiempo en memoria, mientras que las páginas que no se han utilizado recientemente sean las primeras en ser reemplazadas.

3. Detalles de la Implementación:

Procesamiento de Acceso a Páginas:

Se verifica si la página ya está cargada en la lista paginasCargadas.

Si está cargada, el bit de referencia correspondiente se actualiza a true.

Si no está cargada, ocurre una falla de página, se incrementa el contador de fallas y se decide si la página debe ser reemplazada o añadida (si hay espacio).

• Esquema de sincronización usado. Justifique brevemente dónde es necesario usar sincronización y por qué.

La sincronización se emplea en los siguientes métodos

Procesar Acceso , Remplazar Pagina, resetear BitsReferencia

procesarAcceso(String acceso):

Este método accede y modifica varias estructuras compartidas: la lista paginasCargadas, los arreglos referenciaBit[] y modificacionBit[], así como los contadores fallasPagina, accesosExitosos, y totalAccesos.

Razón para sincronización: Como estos recursos pueden ser accedidos simultáneamente por múltiples hilos (por ejemplo, el hilo de procesamiento puede llevar a resultados incorrecto)

reemplazarPagina(int nuevaPagina):

Este método modifica la lista de paginasCargadas y los arreglos de bits de referencia y modificación. Decide qué página debe ser reemplazada basándose en las clases R y M.

Razón para sincronización: Mientras una página está siendo reemplazada, otro hilo no debería modificar la lista paginasCargadas, ya que esto podría corromper los datos o resultar en un comportamiento no deseado. La sincronización garantiza que el reemplazo de páginas se realice de manera atómica, sin interferencias externas.

resetearBitsReferencia():

Este método recorre y modifica el arreglo referenciaBit[], reseteando los bits de referencia periódicamente.

Razón para sincronización: Sincronizar este método asegura que mientras los bits de referencia están siendo reiniciados, ningún otro hilo (especialmente el hilo que procesa accesos) intente leer o modificar los bits de referencia al mismo tiempo, evitando condiciones de carrera que podrían resultar en una incorrecta decisión de reemplazo de página.

• Una tabla con los datos recopilados (y porcentaje de hits y misses por cada escenario simulado).

Parrots\_mod.bpm

Texto

Descripción generada automáticamente con confianza baja

Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Parrots.bpm

Tabla

Descripción generada automáticamente

Tabla

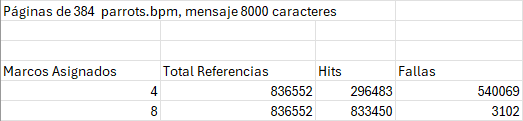
Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente



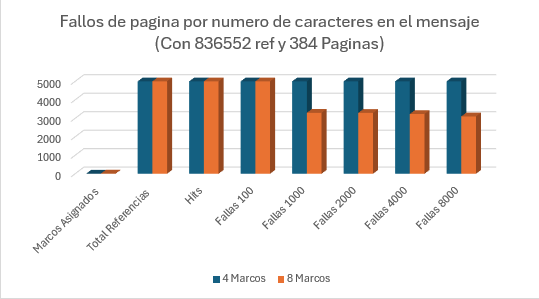
• Una serie de gráficas que ilustren el comportamiento del sistema. Para eso muestre gráficas donde fije tamaño de página y grafique Tamaño de imagen vs. Marcos asignados vs. Porcentaje de hits. La gráfica al final del enunciado ilustra el tipo de gráfica que buscamos.

Caso Parrots\_mod.bpm

Gráfico, Gráfico de barras

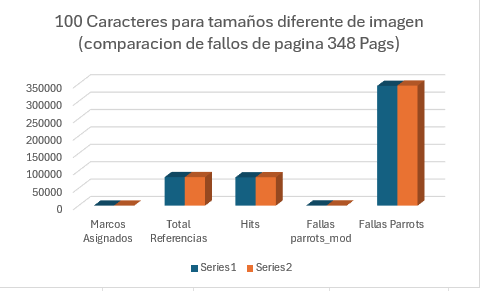
Descripción generada automáticamente

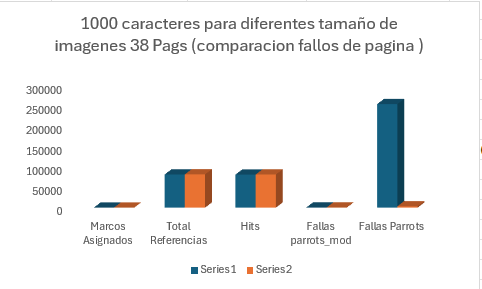
Parrots.bmp

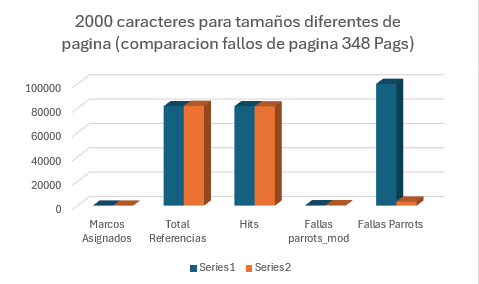


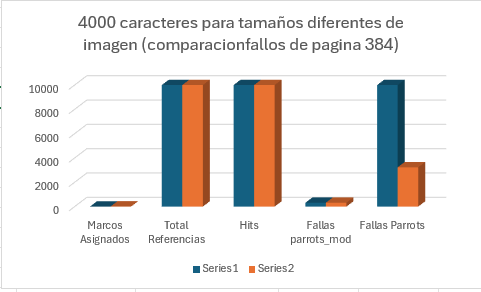
• Corra los escenarios y genere gráficas que muestren los datos recopilados para los diferentes escenarios.

Naranja para 8 marcos y azul para 4 marcos









Gráfico, Gráfico de barras

Descripción generada automáticamente

• Escriba su interpretación de los resultados: ¿corresponden a los resultados que esperaba, con respecto al número de marcos asignados? Explique su respuesta.

En la mayoría de los resultados de parrots\_mod da un comportamiento contrario a lo que se espera ya que se asignan más marcos, pero se incrementan los fallos de página, la hipótesis es porque se utilizó fifo de respaldo si no se encuentra una pagina remplazable en las tres clases del NRU aunque la diferencia entre hits y fallos es mínima. Sin embargo, en el parrots se da un comportamiento esperado y los fallos se reducen de una manera muy significativa cuando se aumentan los marcos de página quizás se da porque los accesos de página son diferentes con relación a los tamaños de los archivos. Aunque se destaca que con el archivo parrots mod los tiempos de espera era máximo dos minutos mientras que los de parrots alcanzaban los 20 minutos

• ¿Si la localidad del problema manejado fuera diferente cómo variarían los resultados? Explique su respuesta. considere una localidad mayor y una localidad menor).

Con una localidad mayor se producirían menos fallos de página ósea al inicio se presentarían, pero después no habrían tantos en cambio con una localidad menor habrían muchos fallos de pagina desde el inicio aunque si se aumentaran los marcos se reduciría un poco pero el impacto será menos pronunciado que en el caso de la localidad mayor, debido a que el conjunto de trabajo es más grande y las páginas necesarias se distribuyen más ampliamente.