

UNIVERSIDAD ESTATAL A DISTANCIA

ESCUELA DE CIENCIAS EXACTAS Y NATURALES

CARRERA INGENIERÍA INFORMÁTICA

CATEDRA DESARROLLO DE SISTEMAS

00881 – Sistemas Operativos

TAREA No 1

Estudiante:

Francisco Campos Sandi

Cédula:114750560

Sede Universitaria:

San Vito

Docente:

Bernarda Delgado Molina

II Cuatrimestre 2024

Contenido

Introducción.....	3
Marco Teórico	4
Memoria Virtual	4
Sistemas Operativos.....	5
Jerarquía de Memoria.....	6
Interrupciones y Manejo de Excepciones.....	7
Desarrollo.....	8
Conclusión	16
Referencias.....	17
Tabla 1: Algoritmos	14
Ilustración 1: Segmentación	11

Introducción

El presente trabajo se centra en la administración de memoria en diferentes sistemas operativos, específicamente Windows, Mainframe y Android. A través de un análisis detallado, se busca comprender cómo cada uno de estos sistemas maneja la memoria para optimizar el rendimiento y la eficiencia. El objetivo es proporcionar una visión completa y comparativa de las técnicas y métodos utilizados en estos entornos, destacando sus características y particularidades.

Se investiga ampliamente sobre los conceptos de paginación y segmentación, dos técnicas fundamentales en la gestión de memoria. Estos conceptos son esenciales para entender cómo los sistemas operativos organizan y gestionan la memoria disponible. La investigación pretende aclarar estos términos y su aplicación práctica en diferentes contextos, brindando una base sólida para el análisis comparativo.

El trabajo también examina la administración de memoria en los procesadores Intel Core. Este análisis es crucial para comprender cómo el hardware interactúa con el software en la gestión de memoria. Se busca detallar las técnicas específicas empleadas por estos procesadores y cómo estas influyen en el rendimiento general del sistema.

Y, se presenta un cuadro comparativo de los algoritmos de reemplazo de páginas, específicamente el algoritmo de reemplazo de páginas menos usadas recientemente (LRU) y el algoritmo de reemplazo de páginas Conjunto de Trabajo. El cuadro incluye las ventajas y desventajas de cada algoritmo, proporcionando una evaluación clara de su efectividad. El trabajo se estructura en un marco teórico, desarrollo, conclusión y referencias, asegurando una presentación organizada y coherente de la investigación.

Marco Teórico

El marco teórico de este trabajo se centra en la comprensión de conceptos fundamentales que son esenciales para analizar la administración de memoria en sistemas operativos y procesadores. Entre estos conceptos se encuentran la memoria virtual, que permite a los sistemas utilizar más memoria de la que físicamente disponen; los sistemas operativos, que son responsables de gestionar los recursos del sistema y proporcionar una interfaz para la ejecución de aplicaciones; la jerarquía de memoria, que organiza los diferentes niveles de almacenamiento desde los más rápidos y costosos hasta los más lentos y económicos; y las interrupciones y manejo de excepciones, que son cruciales para la estabilidad y eficiencia del sistema frente a eventos inesperados y errores.

Memoria Virtual

La memoria virtual es una técnica de gestión de memoria que permite a los sistemas operativos usar más memoria de la que físicamente está disponible en el hardware. Esto se logra mediante la asignación de un espacio de direcciones virtual que se mapea a la memoria física y al almacenamiento secundario (como el disco duro). La memoria virtual facilita la ejecución de programas grandes, mejora la seguridad al aislar los espacios de direcciones de cada proceso, y aumenta la estabilidad del sistema al gestionar eficientemente los recursos de memoria disponibles.

Según Buzón (2020), la memoria permite que los componentes esenciales para el procesamiento y transformación de datos en nuestros ordenadores accedan a la información necesaria; además, la memoria persistente conserva todo nuestro trabajo, recuerdos y pasatiempos dentro de las máquinas (párr. 02).

La memoria virtual se vincula directamente con el tema principal, ya que es una técnica esencial en la administración de memoria en sistemas operativos. Permite que

estos sistemas manejen eficientemente la memoria física y el almacenamiento secundario, facilitando la ejecución de programas grandes y mejorando la estabilidad y seguridad del sistema. La implementación y gestión de la memoria virtual varían entre diferentes sistemas operativos como Windows, Mainframe y Android, lo que hace crucial entender este concepto para analizar cómo cada sistema maneja sus recursos de memoria.

Sistemas Operativos

Un sistema operativo es el software fundamental que gestiona el hardware y los recursos del sistema, proporcionando servicios y funciones esenciales para la ejecución de aplicaciones. Entre sus responsabilidades se encuentra la gestión de la memoria, el procesamiento de tareas, la administración de dispositivos de entrada y salida, y la provisión de una interfaz de usuario. Los sistemas operativos modernos, como Windows, Mainframe y Android, implementan diferentes técnicas de gestión de memoria para optimizar el uso de recursos y mejorar el rendimiento del sistema.

Sy Corvo, (2021) menciona que;

El sistema operativo está compuesto por el software del sistema, que son los programas fundamentales que la computadora necesita para arrancar y funcionar. Cada computadora de escritorio, tableta y teléfono inteligente incluye un sistema operativo para brindar una funcionalidad básica al dispositivo. (párr.02).

Los sistemas operativos son el núcleo de la administración de memoria en cualquier entorno computacional. Ellos son los responsables de implementar y gestionar las técnicas de memoria virtual, paginación, segmentación y los algoritmos de reemplazo de páginas. Cada sistema operativo, ya sea Windows, Mainframe o Android, tiene sus propias estrategias y métodos para optimizar el uso de la memoria y mejorar el

rendimiento del sistema. Comprender cómo funcionan los sistemas operativos es fundamental para analizar las diferencias y similitudes en la administración de memoria entre estos sistemas.

Jerarquía de Memoria

La jerarquía de memoria es una estructura de niveles de almacenamiento que organiza la memoria en un sistema computacional desde la más rápida y costosa (como los registros del procesador y la memoria caché) hasta la más lenta y económica (como el almacenamiento secundario). La jerarquía de memoria incluye:

- **Registros:** Pequeñas cantidades de memoria dentro del procesador para almacenar datos temporales.
- **Memoria Caché:** Memoria de alta velocidad que almacena datos frecuentemente utilizados para acelerar el acceso.
- **Memoria Principal (RAM):** Memoria de acceso aleatorio utilizada para almacenar datos y programas en ejecución.
- **Almacenamiento Secundario:** Dispositivos como discos duros y SSD que proporcionan almacenamiento a largo plazo.

Esta estructura es fundamental para entender cómo los sistemas operativos gestionan y optimizan el acceso a la memoria.

Acuña, (2018) indica que,

El diseño de la jerarquía de memoria en un sistema informático incluye principalmente diferentes dispositivos de almacenamiento. La mayoría de los ordenadores llevan incorporado un almacenamiento adicional para funcionar con más potencia que la capacidad de la memoria principal. Lo siguiente diagrama de

la jerarquía de la memoria es una pirámide jerárquica para la memoria del ordenador. El diseño de la jerarquía de la memoria se divide en dos tipos, como la memoria primaria (interna) y la memoria secundaria (externa). (párr.03).

Este concepto se vincula con el tema principal al proporcionar el contexto necesario para entender cómo los sistemas operativos y procesadores, como los Intel Core, gestionan y optimizan el uso de la memoria en distintos niveles de la jerarquía.

Interrupciones y Manejo de Excepciones

Las interrupciones son señales que alertan al procesador sobre eventos que requieren su atención inmediata, como errores de hardware o solicitudes de servicio de dispositivos periféricos. El manejo de excepciones es el proceso mediante el cual el sistema operativo responde a eventos inesperados o errores, garantizando la estabilidad y continuidad del sistema. En el contexto de la gestión de memoria, las interrupciones pueden ser provocadas por faltas de página o errores de segmentación, y es crucial que el sistema operativo tenga mecanismos eficientes para manejar estas situaciones.

Greyrat, (2022) menciona que;

Cada vez que ocurre una excepción o interrupción, el hardware comienza a ejecutar el código que realiza una acción en respuesta a la excepción. Esta acción puede implicar eliminar un proceso, generar un mensaje de error, comunicarse con un dispositivo externo o colapsar horriblemente todo el sistema informático al iniciar una «Pantalla azul de la muerte» y detener la CPU. Las instrucciones responsables de esta acción residen en el kernel del sistema operativo, y el código que realiza esta acción se denomina código del controlador de interrupciones. Ahora, podemos pensar en el código del controlador como una subrutina del sistema operativo. Luego, después de ejecutar el código del controlador, es

posible continuar la ejecución después de la instrucción donde ocurrió la ejecución o la interrupción. (párr.02).

Por lo que estos conceptos proporcionan una comprensión básica necesaria para abordar de manera efectiva el estudio de la administración de memoria en sistemas operativos y procesadores, estableciendo una base teórica sólida para el análisis posterior de las técnicas específicas de paginación, segmentación y algoritmos de reemplazo de páginas.

Desarrollo

1. ¿Cómo se administra la memoria en los Sistemas Operativos de Windows, de Mainframe y de Android?

La administración de memoria en los sistemas operativos de Windows se basa en un esquema de memoria virtual, donde cada proceso se ejecuta en su propio espacio de direcciones virtuales. Windows utiliza paginación para gestionar la memoria, asignando marcos de página en la memoria física a las páginas virtuales cuando es necesario. Además, emplea técnicas como la compresión de memoria y el archivo de paginación para optimizar el uso de la memoria disponible y mejorar el rendimiento del sistema. El administrador de memoria de Windows también se encarga de la protección de memoria, asegurando que los procesos no interfieran entre sí.

Espinel, (2018) menciona que;

Básicamente, los mainframes son computadoras poderosas con mucha memoria y procesadores que manejan miles de millones de cálculos y transacciones simples en tiempo real. El mainframe es esencial para las bases de datos empresariales, los servidores de eventos y las aplicaciones que requieren flexibilidad, seguridad y agilidad. (párr.03).

En los sistemas operativos de Mainframe, la administración de memoria es mucho más compleja y robusta, debido a la necesidad de manejar grandes volúmenes de datos y múltiples usuarios simultáneamente. Según The QA Testing Channel (2017) un servidor mainframe es una computadora de alto rendimiento diseñada para el procesamiento de datos y la ejecución de tareas a gran escala. Requiere niveles superiores de seguridad y rendimiento en comparación con las computadoras de mediana o pequeña escala. (min 0:22/3:07), Mainframe utiliza técnicas avanzadas como la paginación y la segmentación, además de tener capacidades de memoria compartida para permitir que varios procesos accedan a los mismos datos sin duplicarlos. Estos sistemas también implementan mecanismos de protección y aislamiento de memoria para asegurar que los procesos críticos no sean interrumpidos ni comprometidos, garantizando así la estabilidad y seguridad del entorno operativo.

Android, basado en el kernel de Linux, administra la memoria de manera similar a otros sistemas basados en Unix. Utiliza un sistema de memoria virtual con paginación, permitiendo a las aplicaciones ejecutar en sus propios espacios de direcciones. Android emplea un administrador de memoria específico para dispositivos móviles, que incluye técnicas como la eliminación de aplicaciones en segundo plano cuando la memoria es escasa, para liberar recursos y mantener el rendimiento del dispositivo. Además, Android utiliza una gestión agresiva de caché y una política de prioridad para aplicaciones en primer plano para optimizar el uso de la memoria en un entorno con recursos limitados.

Según Bonilla (2024), los sistemas mainframe están diseñados para realizar un elevado número de operaciones por segundo. Su potencia de procesamiento se evalúa en MIPS (millones de instrucciones por segundo), dado que deben responder a las demandas de millones de usuarios simultáneamente. Estas máquinas suelen incorporar

múltiples procesadores y una capacidad de memoria considerable, lo que les facilita la gestión simultánea de numerosas tareas (párr. 02).

Por lo que, aunque los sistemas operativos de Windows, Mainframe y Android utilizan principios básicos similares para la administración de memoria, como la memoria virtual y la paginación, cada uno adapta estas técnicas a sus necesidades específicas. Windows se enfoca en optimizar el rendimiento de aplicaciones de usuario en un entorno de escritorio, Mainframe gestiona grandes volúmenes de datos y múltiples usuarios en un entorno empresarial, y Android optimiza el uso de memoria para dispositivos móviles con recursos limitados. Estas diferencias reflejan las distintas prioridades y desafíos que enfrenta cada sistema operativo en su contexto de uso.

2. ¿Qué es paginación y qué es segmentación? Investigue ampliamente.

La paginación es una técnica de gestión de memoria utilizada por los sistemas operativos para dividir el espacio de direcciones virtuales de un proceso en bloques de tamaño fijo llamados páginas. Estas páginas se mapean a marcos de página en la memoria física, lo que permite que los procesos se ejecuten sin necesidad de disponer de toda la memoria requerida de manera continua.

Cuando una página no está presente en la memoria física, se produce una falta de página, y el sistema operativo debe cargarla desde el almacenamiento secundario. La paginación evita la fragmentación externa y facilita la asignación de memoria, mejorando la eficiencia y flexibilidad en el uso de los recursos de memoria.

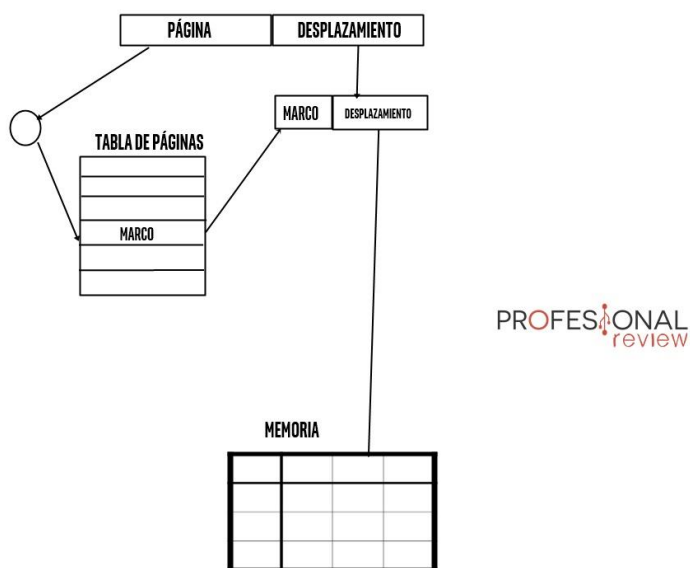
Aller, (2020) indica que;

Se trata de un modelo de organización de memoria física en el que se divide toda la memoria en porciones del mismo tamaño. Esas porciones reciben el nombre de marcos o páginas físicas. Si dividimos la memoria en páginas, podremos

gestionarla mejor. Los marcos se identifican con un número que se denomina «número de página física». A su vez, cada página física se asigna a un proceso de forma exclusiva. Por otro lado, cada proceso tiene un espacio de páginas lógicas, y cada una se mapea o se asocia a un marco. (párr.02).

La segmentación es otra técnica de gestión de memoria que divide el espacio de direcciones de un proceso en segmentos de tamaño variable. Cada segmento representa una unidad lógica, como una función, un objeto o un bloque de datos, y tiene sus propias direcciones de inicio y fin. A diferencia de la paginación, la segmentación permite una mayor organización y protección de la memoria, ya que cada segmento puede tener diferentes permisos de acceso y puede ser ubicado en distintas áreas de la memoria física. Sin embargo, la segmentación puede provocar fragmentación externa, ya que los segmentos son de tamaño variable y pueden dejar huecos no utilizables en la memoria.

Ilustración 1: Segmentación



Tomada de Aller, Á. (2020). *¿Cómo funciona la paginación de memoria?* [Imagen].

Profesional Review. <https://www.profesionalreview.com/2020/06/10/como-funciona-la-paginacion-de-memoria/>

Ambas técnicas, paginación y segmentación, son esenciales para la gestión eficiente de la memoria en sistemas operativos modernos. La paginación es más común en sistemas que requieren flexibilidad y eficiencia en la asignación de memoria, como Windows y Android. En cambio, la segmentación es preferida en entornos que necesitan una mayor protección y organización de la memoria, como los sistemas Mainframe. En muchos casos, los sistemas operativos combinan ambas técnicas para aprovechar sus ventajas y minimizar sus desventajas, proporcionando una gestión de memoria robusta y eficiente.

3. ¿Cómo se administra la memoria en el procesador Intel Core?

La administración de memoria en los procesadores Intel Core se realiza mediante una combinación de hardware y software que trabajan juntos para optimizar el uso de la memoria y mejorar el rendimiento del sistema. En el hardware, los procesadores Intel Core utilizan una jerarquía de memoria que incluye registros, múltiples niveles de caché (L1, L2, y L3), y la memoria principal (RAM). La caché es esencial para la administración de memoria, ya que almacena datos y instrucciones frecuentemente utilizados, reduciendo el tiempo de acceso y mejorando la velocidad de procesamiento.

El procesador Intel Core también implementa una Unidad de Gestión de Memoria (MMU), que traduce las direcciones virtuales generadas por el software a direcciones físicas en la memoria. Esta traducción se realiza utilizando tablas de paginación, permitiendo la implementación de memoria virtual. La MMU también gestiona las faltas de página y otras excepciones relacionadas con la memoria, garantizando que el sistema operativo pueda manejar eficientemente la asignación y liberación de memoria.

Según Castillo (2016), al evaluar la potencia de un procesador, solemos concentrarnos en el número de núcleos y su velocidad, factores que indudablemente

influyen en su rendimiento general. Sin embargo, a menudo se subestima la importancia de la memoria caché, la cual desempeña un papel fundamental. Por ejemplo, contar con una CPU de 6 núcleos acompañada de una caché L3 de 4 o 16 MB puede tener un impacto significativo en el rendimiento, especialmente cuando se manejan múltiples programas simultáneamente. Por lo tanto, es crucial considerar este aspecto al decidir la compra de un procesador, ya que no todo se reduce simplemente a la frecuencia. (párr.10).

En el ámbito del software, los sistemas operativos que funcionan sobre procesadores Intel Core, como Windows, Linux y macOS, utilizan técnicas de paginación para gestionar la memoria virtual. Estos sistemas operativos coordinan con la MMU del procesador para mapear las páginas virtuales a marcos de página en la memoria física, optimizando así el uso de los recursos disponibles. Además, los sistemas operativos implementan algoritmos de reemplazo de páginas para decidir qué páginas deben permanecer en la memoria y cuáles pueden ser transferidas al almacenamiento secundario.

En conjunto, la administración de memoria en los procesadores Intel Core es un proceso sofisticado que involucra tanto el hardware del procesador como el software del sistema operativo. La combinación de una jerarquía de memoria eficiente, una MMU robusta y técnicas avanzadas de paginación permite a los procesadores Intel Core manejar aplicaciones intensivas en memoria de manera eficaz, proporcionando un rendimiento optimizado y una experiencia de usuario fluida.

Tabla 1: Algoritmos

Algoritmo de Reemplazo de Páginas	Ventajas	Desventajas
LRU	<p>Simplicidad: Es fácil de implementar y entender.</p> <p>Buen Rendimiento General: Suele proporcionar buenos resultados en términos de tasa de aciertos (hit rate), especialmente en entornos donde hay buena localidad temporal (páginas que se vuelven a utilizar frecuentemente).</p> <p>Eficacia con Datos Regularmente Accedidos: Funciona bien cuando hay patrones claros de acceso a páginas.</p>	<p>Implementación Costosa: Requiere de una estructura de datos (como listas enlazadas) para rastrear el orden de uso de las páginas, lo cual puede ser costoso en términos de memoria y tiempo de acceso.</p> <p>No Predice Futuro Uso: Puede fallar cuando el patrón de acceso de las páginas no sigue la tendencia "lo menos recientemente usado".</p> <p>Problema de Inversión de Tiempo: Puede resultar en un alto costo computacional para mantener actualizada la lista de acceso reciente.</p>
Conjunto de Trabajo	<p>Adaptable a Cargas de Trabajo Cambiantes:</p>	<p>Complejidad de Implementación: Requiere</p>

	<p>Puede ajustarse a patrones de acceso variables en tiempo real.</p> <p>Optimización del Rendimiento: Puede mejorar el rendimiento general en sistemas donde las cargas de trabajo cambian frecuentemente.</p> <p>Eficiencia en Entornos Dinámicos: Puede adaptarse bien a sistemas con múltiples usuarios o aplicaciones que cambian de prioridad.</p>	<p>de una monitorización continua de las páginas en uso, lo que puede ser complejo de implementar.</p> <p>Posible Overhead: Puede generar overhead adicional al necesitar calcular y ajustar constantemente los conjuntos de trabajo.</p> <p>No Garantiza el Mejor Rendimiento Absoluto: Puede no ser tan efectivo en entornos con patrones de acceso muy predecibles.</p>
--	--	--

Fuente: Elaboración Propia.

Conclusión

En conclusión, la gestión de memoria en sistemas operativos y procesadores es un aspecto crítico para garantizar el rendimiento eficiente y la estabilidad de los sistemas informáticos modernos. La implementación de técnicas como la memoria virtual, la paginación y la segmentación permite a los sistemas operativos como Windows, Mainframe y Android manejar de manera efectiva los recursos de memoria, adaptándose a las demandas variadas de aplicaciones y usuarios. La jerarquía de memoria en procesadores como los Intel Core juega un papel crucial al optimizar el acceso a datos mediante la utilización eficiente de registros, cachés y memoria principal, mejorando así la velocidad de procesamiento y la respuesta del sistema.

Además, los algoritmos de reemplazo de páginas como LRU y Conjunto de Trabajo ofrecen estrategias diferentes para gestionar la memoria física y virtual, cada una con sus ventajas y desventajas particulares. Mientras que LRU se destaca por su simplicidad y buen rendimiento en contextos con buenos patrones de acceso temporal, el Conjunto de Trabajo se adapta mejor a entornos dinámicos donde las cargas de trabajo son cambiantes y necesitan ser gestionadas en tiempo real. La elección del algoritmo adecuado depende del escenario específico y de las prioridades del sistema en términos de eficiencia y rendimiento.

En términos generales, entender y aplicar efectivamente la gestión de memoria es fundamental para optimizar el uso de recursos y mejorar la experiencia del usuario en cualquier sistema informático. La evolución continua de técnicas y tecnologías en este campo refleja la necesidad constante de adaptarse a nuevos desafíos, como el crecimiento de aplicaciones más complejas y exigentes en términos de memoria.

Referencias

- Acuña, M. (2018). *Jerarquía de la memoria en la arquitectura de los ordenadores - electrositio*. Electrositio. <https://electrositio.com/jerarquia-de-la-memoria-en-la-arquitectura-de-los-ordenadores/>
- Aller, Á. (2020, 10 de junio). *¿Cómo funciona la paginación de memoria?* Profesional Review. <https://www.profesionalreview.com/2020/06/10/como-funciona-la-paginacion-de-memoria/>
- Bonilla, L. (2024, 29 de enero). *Qué es un Mainframe, cómo funciona y quién lo utiliza | Data Center Market*. Data Center Market. <https://www.datacentermarket.es/dcm-xl/que-es-un-mainframe-como-funciona-y-quien-lo-utiliza/>
- Buzón, M. (2020, 15 de mayo). *¿Qué es la memoria virtual y para qué sirve?* 🏆 【EXPLICACIÓN】 ⭐. Profesional Review. <https://www.profesionalreview.com/2020/05/15/memoria-virtual/>
- Castillo, J. A. (2016, 2 de mayo). *Qué es la memoria caché L1, L2 y L3 y cómo funciona*. Profesional Review. <https://www.profesionalreview.com/2019/05/02/memoria-cache-l1-l2-y-l3/>
- Espinel, H. (2018). *¿Qué es un mainframe y para qué sirve?* uCloud. <https://ucloudglobal.com/blog/que-es-mainframe/>
- Greyrat, R. (2022, 5 de julio). *Interrupciones y excepciones – barcelona geeks*. Barcelona Geeks – La mayor colección de tutoriales y referencias. <https://barcelonageeks.com/interrupciones-y-excepciones/>
- Sy Corvo, H. (2021, 8 de enero). *Sistema operativo: Para qué sirve, componentes, tipos, ejemplos*. Lifeder. https://www.lifeder.com/sistema-operativo/#google_vignette

The QA Testing Channel. (2017, 2 de julio). *¿Qué es Mainframe? en 3 minutos* [Video].

YouTube. <https://www.youtube.com/watch?v=k-skEZc-Tl8>