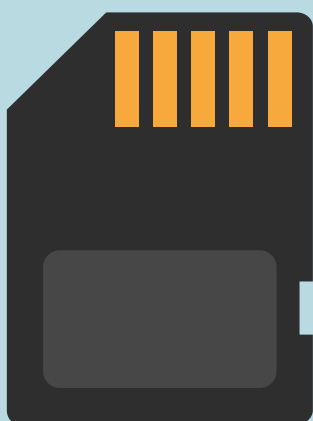


06 DE DICIEMBRE

ADMINISTRACIÓN DE MEMORIA

Elaborado por:

Jimena Mendoza Martinez Angie Ximena Olvera
Raymundo Said Moreno Tomas Alexandro Moreno
Jinathan Guadalupe Guevara



¿QUE ES?

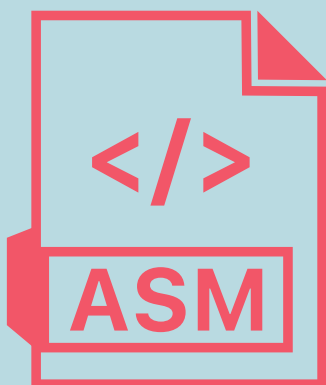
La administración de memoria es crucial para el desempeño de los programas.

Su mal uso impacta directamente el rendimiento del sistema.

IMPORTANCIA

La administración de memoria impacta directamente el desempeño del sistema.

Un mal uso genera errores y ralentiza los procesos.



COMPARACIÓN: ENSAMBLADORES VS. COMPILADORES

- Ensambladores: Gestión de memoria limitada, manual y propensa a errores.
- Compiladores: Utilizan estrategias avanzadas para optimizar el uso de memoria.

EVOLUCIÓN EN LENGUAJES DE PROGRAMACIÓN

- Lenguajes antiguos: Uso no controlado de punteros, causando fallas comunes.
- Lenguajes modernos: Implementan recolectores de basura que eliminan memoria no utilizada. Optimizan el rendimiento general del sistema.



ESTRUCTURA

- Organización como un arreglo lineal de localidades de 1 byte.
- Cada localidad tiene una dirección única que la identifica.

TECNICAS

- Segmentación: División de la memoria en segmentos lógicos según el programa (código, datos, pila).
- Paginación: División en bloques de tamaño fijo (páginas), asignados a espacios de memoria llamados marcos.



Referencias: Material proporcionado por la maestra Thelma Carrillo

06 DE DICIEMBRE

PROPÓSITOS Y ESTRATEGIAS DE LA ADMINISTRACIÓN DE MEMORIA

Elaborado por:

Jimena Mendoza Martinez Angie Ximena Olvera
Raymundo Said Moreno Tomas Alexandro Moreno
Jinathan Guadalupe Guevara

PROPÓSITOS PRINCIPALES

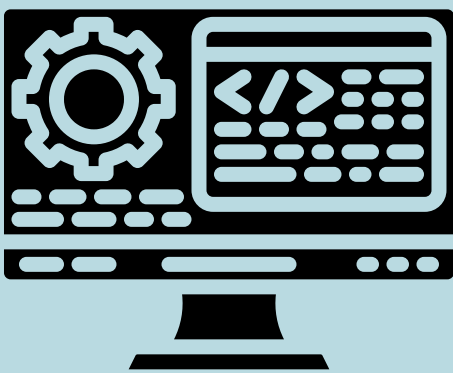


1. PROTECCIÓN

Si varios programas comparten la memoria principal, se debería asegurar que el programa no sea capaz de cambiar las ubicaciones no pertenecientes a él.

2. COMPORTAMIENTO

Este objetivo parece contradecir al anterior, sin embargo, a veces es necesario para los usuarios poder compartir y actualizar información (por ejemplo, en una base de datos) y, si se organiza la tarea de entrada a la misma, se puede evitar el tener varias copias de la rutina.

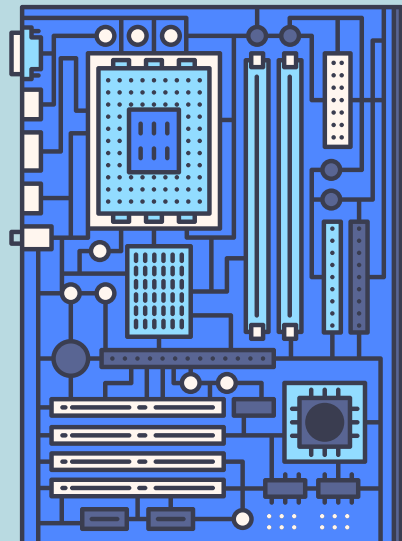


3. REUBICACIÓN

- Permite que varios programas ocupen memoria al mismo tiempo.
- Usa direccionamiento relativo en lugar de absoluto, adaptándose dinámicamente.

4. ORGANIZACIÓN FÍSICA Y LÓGICA

- Física: Combina memoria principal (rápida y cara) con memoria secundaria (lenta pero económica).
- Lógica: Alinea el uso de procedimientos, funciones, subrutinas y arreglos en un espacio secuencial de memoria.



PROBLEMAS COMUNES Y SOLUCIONES



PROBLEMAS

- Uso descontrolado de punteros.
- Acceso indebido entre programas.

SOLUCIONES

- Lenguajes modernos : Implementan restricciones para los punteros.
- Recolectores de basura : Liberan memoria innecesaria automáticamente.



Referencias: Material proporcionado por la maestra Thelma Carrillo