

EJERCICIOS TEMA 9: EL SISTEMA DE ARCHIVOS

1. ¿Qué funciones realiza el gestor de archivos?

El gestor de archivos realiza las siguientes funciones principales:

Creación y eliminación de archivos

Creación y eliminación de directorios

Manipulación de archivos (abrir, cerrar, leer, escribir)

Gestión del espacio libre en disco

Control de acceso y permisos

Organización jerárquica de archivos y directorios

Mantenimiento de la integridad de los datos

2. ¿Qué es un archivo?

Un archivo es una unidad lógica de almacenamiento que contiene información relacionada (datos, programas, etc.). Es un conjunto de bytes almacenados en un dispositivo de almacenamiento secundario, que tiene un nombre, atributos y permisos asociados.

3. ¿A qué se denomina comúnmente metadatos?

Los metadatos son datos que describen otros datos. En sistemas de archivos se refieren a:

Nombre del archivo

Tamaño

Fechas (creación, modificación, acceso)

Permisos y atributos

Ubicación física en el disco

Tipo de archivo

Propietario

4. Indica la diferencia entre fragmentación interna y fragmentación Externa

Fragmentación interna: Ocurre cuando se asigna más espacio del necesario a un archivo, dejando parte del bloque sin usar. Es típica en asignación contigua o con bloques de tamaño fijo.

Fragmentación externa: Ocurre cuando el espacio libre en disco está dividido en pequeños fragmentos dispersos, haciendo difícil asignar espacio contiguo para nuevos archivos aunque haya suficiente espacio total libre.

5. Haz una tabla comparativa sobre las diferentes formas de gestión del espacio libre en disco, indicando en qué consiste cada procesos, las ventajas que presenta y los inconvenientes

Método	Descripción	Ventajas	Inconvenientes
Bitmap	Mapa de bits donde cada bit representa un bloque (1=ocupado, 0=libre)	Simple de implementar, fácil encontrar bloques libres consecutivos	Requiere espacio adicional, puede ser ineficiente para discos muy grandes
Lista enlazada	Bloques libres se enlazan mediante punteros	No requiere espacio adicional, eficiente en uso de espacio	Difícil encontrar bloques libres consecutivos, acceso secuencial
Agrupación	Combina lista enlazada con bitmap, almacena direcciones de bloques libres	Más rápido para encontrar bloques libres, reduce fragmentación	Mayor complejidad de implementación
Conteo	Almacena dirección del primer bloque libre y número de bloques libres	Eficiente para áreas grandes de bloques libres	Menos eficiente cuando hay muchos pequeños huecos

6. Explica mediante un ejemplo la asignación contigua en la asignación de archivos, indicando la estrategia usada a la hora de asignar el espacio en disco

Ejemplo: Supongamos un disco con bloques numerados del 0 al 99. Un archivo A necesita 5 bloques. El sistema busca espacio contiguo libre y lo asigna (por ejemplo bloques 20-24).

Estrategias comunes:

First-fit: Asigna el primer espacio contiguo suficiente que encuentre

Best-fit: Asigna el espacio contiguo más pequeño que sea suficiente

Worst-fit: Asigna el espacio contiguo más grande disponible

7. ¿En qué consiste la asignación enlazada? ¿En qué difiere de la asignación contigua?

Asignación enlazada: Cada bloque del archivo contiene un puntero al siguiente bloque. Los bloques pueden estar dispersos por el disco.

Diferencias:

En contigua los bloques son adyacentes, en enlazada pueden estar en cualquier lugar

Enlazada no sufre fragmentación externa

Contigua permite acceso directo, enlazada solo acceso secuencial

Enlazada necesita espacio adicional para punteros

8. ¿Por qué no hay fragmentación externa en la asignación contigua?

Porque los bloques pueden estar en cualquier ubicación del disco y no necesitan ser contiguos. Cada bloque nuevo puede asignarse en cualquier espacio libre disponible, sin importar su ubicación física.

9. ¿Qué ventajas e inconvenientes presenta la tabla en memoria en asignación enlazada?

Ventajas:

Acceso más rápido a los bloques (no hay que seguir punteros en disco)

Mayor confiabilidad (si se pierde un puntero en disco, se puede recuperar de la tabla en memoria)

Inconvenientes:

Consume memoria principal

Para discos muy grandes, la tabla puede ser demasiado grande

En caso de fallo del sistema, puede perderse información si no está sincronizada con el disco

10. ¿Qué es el bloque de índice? ¿Dónde se usa? ¿Para qué?

El bloque de índice es una estructura que contiene punteros a todos los bloques que componen un archivo. Se usa en la asignación indexada.

Para qué sirve:

Permite acceso directo a cualquier bloque del archivo

Elimina la necesidad de recorrer una cadena de punteros

Facilita la gestión de archivos grandes

11. Si el bloque de índice es demasiado pequeño, no podrá almacenar todos los punteros para un archivo grande, ¿qué mecanismos se usan para tratar esta situación? Explícalos

Cuando un archivo es demasiado grande para un solo bloque de índice se usan:

Índice multinivel: Se crean bloques de índice de segundo nivel que apuntan a bloques de índice de primer nivel

Índice combinado: Algunos punteros apuntan directamente a bloques de datos y otros a bloques de índice

Cadenas de bloques de índice: Similar a asignación enlazada, pero para bloques de índice

12. ¿Cuáles son los sistemas de archivos más conocidos?

FAT (FAT12, FAT16, FAT32)

NTFS (Windows)

ext2, ext3, ext4 (Linux)

HFS+ (macOS)

APFS (macOS moderno)

exFAT

ZFS

Btrfs

13. Explica la estructura del sistema de archivos FAT

La FAT (File Allocation Table) tiene:

Sector de arranque: Contiene información básica del sistema

Tabla FAT: Es una tabla con entradas para cada cluster del disco, indicando:

Cluster libre

Cluster ocupado (con puntero al siguiente cluster del archivo)

Cluster defectuoso

Último cluster de un archivo

Copia de la FAT: Para redundancia

Directorio raíz

Área de datos: Donde se almacenan los archivos y subdirectorios

14. Si tenemos un sistema FAT32, ¿cuál es su capacidad de gestión?

FAT32 puede manejar:

Tamaño máximo de archivo: 4 GB - 1 byte ($2^{32} - 1$ bytes)

Tamaño máximo de volumen: teóricamente hasta 2 TB, pero en práctica suele limitarse a 32 GB para mejor rendimiento

Número máximo de archivos: limitado por el tamaño del directorio raíz y espacio disponible

15. ¿Qué es el directorio raíz?

El directorio raíz es el directorio principal del sistema de archivos, del cual cuelgan todos los demás directorios y archivos. En algunos sistemas (como FAT) tiene una ubicación fija y tamaño limitado, mientras que en otros (como NTFS o ext) puede crecer dinámicamente.

16. Indica las características de los sistemas de archivos NTFS

Soporte para archivos grandes (hasta 16 EB teóricos)

Journaling para mayor integridad ante fallos

Compresión y cifrado transparente

Soporte para permisos ACL avanzados

Cuotas de disco

Puntos de repaseo (para montar volúmenes, symbolic links)

Soporte para nombres largos y caracteres Unicode

Tolerancia a fallos mediante redundancia de metadatos

Sparse files (archivos con huecos)

Transacciones (en versiones recientes)

17. ¿Qué es una base de datos relacional? ¿Por qué está compuesta?

Una base de datos relacional es un conjunto de datos organizados en tablas (relaciones) con filas (tuplas) y columnas (atributos), donde las relaciones entre datos se establecen mediante claves.

Componentes:

Tablas con relaciones entre ellas

Claves primarias y foráneas

Restricciones de integridad

Lenguaje SQL para consultas

¿Para qué sirve?

Para qué sirve:

Almacenar y organizar grandes cantidades de datos

Garantizar integridad y consistencia de los datos

Permitir consultas complejas y relaciones entre datos

Facilitar el acceso concurrente a los datos

18. Indica las diferencias entre ext2, ext3 y ext4

El ext2 no contiene Journaling en cambio el ext3 y ext4 sí, el tamaño máximo de ext4 es superior al del ext2 y ext3 teóricamente, si pasamos a los subdirectorios tendremos 32.000 en el ext2 y ext3 e ilimitados en el ext4.

19. ¿Qué es el superbloque y qué información tiene?

El superbloque es una estructura crítica que contiene información global sobre el sistema de archivos. Contiene:

Tamaño del sistema de archivos

Número de bloques libres/ocupados

Tamaño de los bloques

Tipo de sistema de archivos

Información de montaje (última vez montado, estado limpio/sucio)

Puntero a la lista de bloques libres

Información del journal (en sistemas con journaling)

20. Define:

a) Descriptor de grupo: En sistemas ext, describe un grupo de bloques

b) Inodo: Estructura de datos en sistemas Unix que almacena metadatos de un archivo

c) TRIM: Comando que informa al SSD qué bloques ya no están en uso y pueden ser borrados internamente