

Trabajo Práctico Nº 6Administración de E/S – Administración de Discos
Administración de Archivos**Administración de E/S****1.- Dispositivos**

a) Los dispositivos, según la forma de transferir los datos, se pueden clasificar en 2 tipos:

- Ø Orientados a bloques
- Ø Orientados a flujos

Describa las diferencias entre ambos tipos.

b) Cite ejemplos de dispositivos de ambos tipos.

c) Enuncie las diferencias que existen entre los dispositivos de E/S y que el SO debe considerar.

2.- Técnicas de E/S

Describa como trabajan las siguientes técnicas de E/S

- Ø E/S programada
- Ø E/S dirigida por interrupciones
- Ø DMA (Acceso Directo a Memoria)

3.- La tecnica de E/S programa puede trabajar de dos formas:

- Ø E/S mapeada
- Ø E/S aislada

Indique como trabajan estas 2 técnicas.

4.- Enuncie las metas que debe perseguir un SO para la administración de la entrada salida.**5.- Drivers**

- a) ¿Qué son?
- b) ¿Qué funciones mínimas deben proveer?
- c) ¿Quién determina cuales deben ser estas funciones?

6.- Realice un grafico que marque la relación entre el Subsistema de E/S, los drivers, los controladores de dispositivos y los dispositivos.

7.- Describa mediante un ejemplo los pasos mínimos que se suceden desde que un proceso genera un requerimiento de E/S hasta que el mismo llega al dispositivo.

8.- Enuncie que servicios provee el SO para la administración de E/S**Administración de Discos**

9.- Describa en forma sintética, cómo es la organización física de un disco, puede utilizar gráficos para mayor claridad.

10. La velocidad promedio para la obtención de datos de un disco esta dada por la suma de los siguientes tiempos:

- Ø *Seek Time.*
- Ø *Latency Time.*
- Ø *Transfer Time.*

De una definición para estos tres tiempos.



11.- Suponga un disco con las siguientes características:

- Ø 7 platos con 2 caras utilizables cada uno.
- Ø 1100 cilindros
- Ø 300 sectores por pista, donde cada sector de es 512 bytes.
- Ø Seek Time de 10 ms
- Ø 9000 RPM.
- Ø Velocidad de Transferencia de 10 MiB/s (Mebibyte por segundos).

- a) Calcule la capacidad total del disco.
- b) ¿Cuántos sectores ocuparía un archivo de tamaño de 3 MiB (Mebibytes)?
- c) Calcule el tiempo de transferencia real de un archivo de 15 MiB (Mebibytes). grabado en el disco de manera secuencial (todos sus bloques almacenados de manera consecutiva)
- d) Calcule el tiempo de transferencia real de un archivo de 16 MiB (Mebibytes). grabado en el disco de manera aleatoria.

12.- El Seek Time es el parámetro que posee mayor influencia en el tiempo real necesario para transferir datos desde o hacia un disco. Es importante que el SO planifique los diferentes requerimientos que al disco para minimizar el movimiento de la cabeza lecto-grabadora. Analicemos las diferentes políticas de planificación de requerimientos a disco con un ejemplo:

Supongamos un *Head* con movimiento en 200 *tracks* (numerados de 0 a 199), que está en el track 83 atendiendo un requerimiento y anteriormente atendió un requerimiento en el *track* 75.

Si la cola de requerimientos es: 86, 147, 91, 177, 94, 150, 102, 175, 130, 32, 120, 58, 66, 115. Realice los diagramas para calcular el total de movimientos de *head* para satisfacer estos requerimientos de acuerdo a los siguientes algoritmos de *scheduling* de discos.

- | | |
|---|------------------------------------|
| a) FCFS (<i>First Come, First Served</i>) | d) Look |
| b) SSTF (<i>Shortest Seek Time First</i>) | e) C-Scan (<i>Circular Scan</i>) |
| c) Scan | f) C-Look (<i>Circular Look</i>) |

13.- ¿Alguno de los algoritmos analizados en el ejercicio anterior pueden causar inanición de requerimientos?

14.- Supongamos un *Head* con movimiento en 300 pistas (numerados de 0 a 299), que esta en la pista 143 atendiendo un requerimiento y anteriormente atendió un requerimiento en la pista 125.

Si la cola de requerimientos es: 126, 147, 81, 277, 94, 150, 212, 175, 140, 225, 280, 50, 99, 118, 22, 55; y después de 30 movimientos se incorporan los requerimientos de las pistas 75, 115, 220 y 266. Realice los diagramas para calcular el total de movimientos de *head* para satisfacer estos requerimientos de acuerdo a los siguientes algoritmos de *scheduling* de discos:

- | | |
|---------|-----------|
| a) FCFS | d) Look |
| b) SSTF | e) C-Scan |
| c) Scan | f) C-Look |

15.- Supongamos un *Head* con movimiento en 300 pistas (numerados de 0 a 299), que esta en la pista 140 atendiendo un requerimiento y anteriormente atendió un requerimiento en la pista 135.

Si la cola de requerimientos es: 99, 110, 42, 25, 186, 270, 50, 99, 147^{PF}, 81, 257, 94, 133, 212, 175, 130; y después de 30 movimientos se incorporan los requerimientos de las pistas 85, 150^{PF}, 202 y 288; y después de otros 40 movimientos más se incorporan los requerimientos de las pistas 75, 149^{PF}, 285, 201 y 59. Realice los diagramas para calcular el total de movimientos de *head* para satisfacer estos requerimientos de acuerdo a los siguientes algoritmos de *scheduling* de discos:

- | | |
|---------|---------|
| a) FCFS | c) Scan |
| b) SSTF | d) Look |



e) C-Scan

f) C-Look

Administración de Archivos**16.-** Dados los siguientes métodos de administración de espacio de un archivo:

- Ø Asignación contigua
- Ø Asignación enlazada
- Ø Asignación indexada

- a) Describa como trabaja cada uno
- b) Cite ventajas y desventajas de cada uno.

17.- Gestión de espacio libre:

Dados los siguientes métodos de gestión de espacio libre en un disco:

- Ø Tabla de bits.
- Ø Lista Ligada
- Ø Agrupamiento
- Ø Recuento

- a) Describa como trabajan estos métodos
- b) Cite ventajas y desventajas de cada uno.

18.- Ejemplo: Gestión de archivos en UNIX.

El sistema de archivos de UNIX utiliza una versión modificada del esquema de Asignación Indexada para la administración de espacio de los archivos.

Cada archivo o directorio esta representado por una estructura que mantiene, entre otra información, las direcciones de los bloques que contienen los datos del archivo: el **I-NODO**.

Cada I-NODO contiene 13 direcciones a los bloques de datos, organizadas de la siguiente manera:

- Ø 10 de direccionamiento directo.
- Ø 1 de direccionamiento indirecto simple
- Ø 1 de direccionamiento indirecto doble
- Ø 1 de direccionamiento indirecto triple.

- a) Realice un grafico que describa la estructura del I-NODO y de los bloques de datos.

Cada bloque es de 1 Kib (Kibibit). Si cada dirección para referenciar un bloque es de 32 bits:

- b) ¿Cuántas referencias (direcciones) a bloque pueden contener un bloque de disco?
- c) ¿Cuál sería el tamaño máximo de un archivo?

19. Analice las siguientes fórmulas necesarias para localizar un I-NODO en la lista de *inodos*:
$$\text{nro bloque} = ((\text{nro de inodo} - 1) / \text{nro. de inodos por bloque}) + \text{bloque de comienzo de la lista de inodos.}$$

$$\text{Desplazamiento del inodo en el bloque} = ((\text{nro de inodo} - 1) \bmod (\text{número de inodos por bloque})) * \text{medida de inodo del disco}$$

- a) Según la primer fórmula, asumiendo que en el bloque 2 está en el comienzo de la lista de *inodos* y que hay 8 *inodos* por bloque: calcule donde se encuentra el *inodo* 8 y el 9. ¿Dónde estarían para bloque de disco de 16 *inodos*?
- b) De acuerdo a la segunda fórmula, si cada *inodo* del disco ocupa 64 bytes y hay 8 inodos por bloque de disco, el *inodo* 8 comienza en el desplazamiento 448 del bloque de disco. ¿Dónde empieza el 6? Si fueran *inodos* de 128 bytes y 24 *inodos* por bloque: ¿dónde empezaría el *inodo* 8?

