

## **Conceptos de Sistemas Operativos (IC)**

### **Introducción a los Sistemas Operativos (LI, LS, APU, ATIC)**

#### **Trabajo Práctico N° 4**

##### **Objetivo**

Continuar los temas relacionados a la administración de memoria, introduciendo paginación por demanda y su utilización para implementar memoria virtual, analizando el impacto en la ejecución de los procesos.

Analizar el componente de Entrada/Salida de un Kernel, su organización, objetivos y funcionalidades en la administración de los dispositivos. Estudiar la implementación de Sistemas de Archivos. Estudiar el impacto de la utilización de Cache de Disco.

##### **Temas incluidos**

Paginación por demanda. Algoritmos de reemplazos. Memoria Virtual. Entrada-Salida. Dispositivos. Sistemas de Archivos. Buffer Cache.

1. Memoria Virtual
  - a. Describa qué beneficios introduce este esquema de administración de la memoria.
  - b. ¿En que se debe apoyar el SO para su implementación?
  - c. Al implementar esta técnica utilizando paginación por demanda, las tablas de páginas de un proceso deben contar con información adicional además del marco donde se encuentra la página. ¿Cuál es esta información? ¿Por qué es necesaria?
2. Fallos de Página (Page Faults):
  - a. ¿Cuándo se producen?
  - b. ¿Quién es responsable de detectar un fallo de página?
  - c. Describa las acciones que emprende el Kernel cuando se produce un fallo de página.
3. Direcciones. Si se dispone de un espacio de direcciones virtuales de 32 bits, donde cada dirección referencia 1 byte:
  - a. ¿Cuál es el tamaño máximo del espacio de direcciones de un proceso?
  - b. Si el tamaño de página es de 512Kb. ¿Cuál es el número máximo de páginas que puede tener un proceso?
  - c. Si el tamaño de página es de 512Kb. y se disponen de 256 Mb. de memoria principal ¿Cuál es el número de marcos que puede haber?
  - d. Si se utilizaran 2 Kb. para cada entrada en la tabla de páginas de un proceso: ¿Cuál sería el tamaño máximo de la tabla de páginas de cada proceso?

4. Como se vio en el ejercicio anterior, la tabla de páginas de un proceso puede alcanzar un tamaño considerablemente grande..Existen varios enfoques para administrar las tablas de páginas:

- ☐ Tablas de páginas de 1 nivel.
- ☐ Tablas de páginas de 2 niveles o mas
- ☐ Tablas de páginas invertidas.

Explique brevemente cómo trabajan estos mecanismos. Tenga en cuenta analizar cómo se resuelven las direcciones y qué beneficios presentan una sobre otra.

5. Suponga que la tabla de páginas para un proceso que se está ejecutando es la que se muestra a continuación:

Página	Bit V	Bit R	Bit M	Marco
0	1	1	0	4
1	1	1	1	7
2	0	0	0	-
3	1	0	0	2
4	0	0	0	-
5	1	0	1	0

Asumiendo que:

- ☐ El tamaño de la página es de 512 bytes
- ☐ Cada dirección de memoria referencia 1 byte
- ☐ Los marcos se encuentran contiguos y en orden en memoria (0, 1, 2.. ) a partir de la dirección física 0.

¿Qué dirección física, si existe, correspondería a cada una de las siguientes direcciones virtuales? (No gestionar ningún fallo de página en caso de producirse)

- a. 1052
- b. 2221
- c. 5499
- d. 3101

6. Tamaño de la Página. La selección del tamaño de la página influye de manera directa sobre el funcionamiento de la memoria virtual. Compare las siguientes situaciones con respecto al tamaño de página, indicando ventajas y desventajas:

- ☐ Un tamaño de página pequeño.
- ☐ Un tamaño de página grande.

7. Asignación de marcos a un proceso (Conjunto de trabajo o Working Set). Con la memoria virtual paginada, no se requiere que todas las páginas de un proceso se encuentren en memoria. El SO debe controlar cuantas páginas de un proceso puede tener en la memoria principal. Existen 2 políticas que se pueden utilizar:

- ☐ Asignación Fija
- ☐ Asignación Dinámica.

- a. Describa cómo trabajan estas 2 políticas.
- b. Dada la siguiente tabla de procesos y las páginas que ellos ocupan, y

teniéndose 40 marcos en la memoria principal, cuántos marcos le corresponderá a cada proceso si se usa la técnica de Asignación Fija:

- Reparto Equitativo
- Reparto Proporcional

Proceso	Total de Páginas Usadas
1	15
2	20
3	20
4	8

c. ¿Cuál de los 2 repartos usados en b) resultó más eficiente? Justifique

8. Reemplazo de páginas (selección de una víctima). ¿Qué sucede cuando todos los marcos en la memoria principal están usados por las páginas de los procesos y se produce un fallo de página? El SO debe seleccionar una de las páginas que se encuentra en memoria principal (en un marco) como víctima, la que será reemplazada por la nueva página que produjo el fallo.

Considere los siguientes algoritmos de selección de víctimas básicos:

LRU

FIFO

☐ OPT (Óptimo)

☐ Segunda Chance

- a. Clasifique estos algoritmos de malo a bueno de acuerdo a la tasa de fallos de página que se obtienen al utilizarlos.
- b. Analice su funcionamiento. ¿Cómo se podrían implementar?
- c. Sabemos que la página a ser reemplazada puede estar modificada. ¿Cómo detecta el Kernel esta situación? ¿Qué acciones adicionales deben realizarse cuando se encuentra ante esta situación?

9. Alcance del reemplazo. Al momento de tener que seleccionar una página víctima, el Kernel puede optar por 2 políticas a utilizar:

☐ Reemplazo local

☐ Reemplazo global

- a. Describa cómo trabajan estas 2 políticas.
- b. ¿Es posible utilizar la política de "Asignación Fija" de marcos junto con la política de "Reemplazo Global"? Justifique.

10. Considere la siguiente secuencia de referencias de páginas:

1, 2, 15, 4, 6, 2, 1, 5, 6, 10, 4, 6, 7, 9, 1, 6, 12, 11, 12, 2, 3, 1, 8, 1, 13, 14, 15, 3, 8

- a. Si se disponen de 5 marcos. ¿Cuántos fallos de página se producirán si se utilizan las siguientes técnicas de selección de víctima? (Considere una política de Asignación Dinámica y Reemplazo Global)
  - i. Segunda Chance

- ii. FIFO
- iii. LRU
- iv. OPT

- b. Suponiendo que cada atención de un fallo se página requiere de 0,1 seg. Calcular el tiempo consumido por atención a los fallos de páginas para los algoritmos de a).

11. Sean los procesos A, B y C tales que necesitan para su ejecución las siguientes páginas:

- ☐ **A:** 1, 3, 1, 2, 4, 1, 5, 1, 4, 7, 9, 4
- ☐ **B:** 2, 4, 6, 2, 4, 1, 8, 3, 1, 8
- ☐ **C:** 1, 2, 4, 8, 6, 1, 4, 1

Si la secuencia de ejecución es tal que los procesos se ejecutan en la siguientes secuencia:

- |                               |                                |
|-------------------------------|--------------------------------|
| 1. <b>B</b> demanda 2 páginas | 8. <b>C</b> demanda 4 páginas  |
| 2. <b>A</b> demanda 3 páginas | 9. <b>A</b> demanda 3 páginas  |
| 3. <b>C</b> demanda 2 páginas | 10. <b>B</b> demanda 3 páginas |
| 4. <b>B</b> demanda 3 páginas | 11. <b>C</b> termina           |
| 5. <b>A</b> demanda 3 páginas | 12. <b>A</b> demanda 3 páginas |
| 6. <b>C</b> demanda 2 páginas | 13. <b>B</b> termina           |
| 7. <b>B</b> demanda 2 páginas | 14. <b>A</b> termina           |

- a. Considerando una política de Asignación Dinámica y Reemplazo Global y disponiéndose de 7 marcos. ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de víctimas:

- i. LRU
- ii. Segunda Chance

- b. Considerando una política de Asignación Fija con reparto equitativo y Reemplazo Local y disponiéndose de 9 marcos. ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de víctimas:

- i. LRU
- ii. Segunda Chance

12. Sean los procesos A, B y C tales que necesitan para su ejecución las siguientes páginas:

- ☐ **A:** 1, 2, 1, 7, 2, 7, 3, 2
- ☐ **B:** 1, 2, 5, 2, 1, 4, 5
- ☐ **C:** 1, 3, 5, 1, 4, 2, 3

Si la secuencia de ejecución es tal que los procesos se ejecutan en la siguiente manera:

- |                               |                              |
|-------------------------------|------------------------------|
| 1. <b>C</b> demanda 1 página  | 3. <b>C</b> demanda 1 página |
| 2. <b>A</b> demanda 2 páginas | 4. <b>B</b> demanda 1 página |

- |                                   |                                |
|-----------------------------------|--------------------------------|
| 5. <b>A</b> demanda 1 página      | 14. <b>B</b> demanda 2 páginas |
| 6. <b>C</b> modifica la página 1  | 15. <b>C</b> demanda 2 páginas |
| 7. <b>B</b> demanda 2 páginas     | 16. <b>C</b> demanda 1 página  |
| 8. <b>A</b> demanda 1 página      | 17. <b>A</b> demanda 1 página  |
| 9. <b>C</b> demanda 1 página      | 18. <b>B</b> termina           |
| 10. <b>B</b> modifica la página 2 | 19. <b>A</b> demanda 2 páginas |
| 11. <b>A</b> modifica la página 2 | 20. <b>C</b> demanda 1 página  |
| 12. <b>B</b> demanda 2 páginas    | 21. <b>A</b> termina           |
| 13. <b>A</b> demanda 1 página     | 22. <b>C</b> termina           |

Considerando una política de Asignación Dinámica y Reemplazo Global y disponiéndose de 7 marcos, debiéndose guardar 1 marco para la gestión de descarga asincrónica de páginas modificadas ¿Cuántos fallos de página se producirán si se utiliza la técnica de selección de víctima:

- Segunda Chance
- FIFO
- LRU

13. Hiperpaginación (Trashing)

- ¿Qué es?
- ¿Cuáles pueden ser los motivos que la causan?
- ¿Cómo la detecta el SO?
- Una vez que lo detecta, ¿qué acciones puede tomar el SO para eliminar este problema?

14. Analice y compare las técnicas de PFF y del Working Set. ¿Como las mismas permiten determinar la existencia de trashing?

15. Considere un sistema cuya memoria principal se administra mediante la técnica de paginación por demanda que utiliza un dispositivo de paginación, algoritmo de reemplazo global LRU y una política de asignación que reparte marcos equitativamente entre los procesos. El nivel de multiprogramación es actualmente, de 4.

Ante las siguientes mediciones:

- Uso de CPU del 13%, uso del dispositivo de paginación del 97%.
- Uso de CPU del 87%, uso del dispositivo de paginación del 3%.
- Uso de CPU del 13%, uso del dispositivo de paginación del 3%.

Analizar para cada caso:

- ☐ ¿Qué sucede?
- ☐ ¿Puede incrementarse el nivel de multiprogramación para aumentar el uso de la CPU?
- ☐ ¿La paginación está siendo útil para mejorar el rendimiento del sistema?

16. Considere un sistema cuya memoria principal se administra mediante la técnica de paginación por demanda. Considere las siguientes medidas de utilización:

- ☐ Utilización del procesador: 20%
- ☐ Utilización del dispositivo de paginación: 97,7%
- ☐ Utilización de otros dispositivos de E/S: 5%

Cuales de las siguientes acciones pueden mejorar la utilización del procesador:

- a) Instalar un procesador más rápido
- b) Instalar un dispositivo de paginación mayor
- c) Incrementar el grado de multiprogramación
- d) Instalar mas memoria principal
- e) Decrementar el quantum para cada proceso

17. La siguiente fórmula describe el tiempo de acceso efectivo a la memoria al utilizar paginación para la implementación de la memoria virtual:

$$TAE = At + (1 - p) * Am + p * (Tf Am)$$

Donde:

- TAE = tiempo de acceso efectivo
- p = tasa de fallo de página ( $0 \leq p \leq 1$ )
- Am = tiempo de acceso a la memoria real
- Tf = tiempo se atención de una fallo de pagina
- At = tiempo de acceso a la tabla de páginas. Es igual al tiempo de acceso a la memoria (Am) si la entrada de la tabla de páginas no se encuentra en la TLB.

Suponga que tenemos una memoria virtual paginada, con tabla de páginas de 1 nivel, y donde la tabla de páginas se encuentra completamente en la memoria.

Servir una falla de página tarda 300 nanosegundos si hay disponible un marco vacío o si la página reemplazada no se ha modificado, y 500 nanosegundos si se ha modificado. El tiempo de acceso a memoria es de 20 nanosegundos y el de acceso a la TLB es de 1 nanosegundo

- a. Si suponemos una tasa de fallos de página de 0,3 y que siempre contamos con un marco libre para atender el fallo ¿Cuál será el TAE si el 50% de las veces la entrada de la tabla de páginas se encuentra en la TLB (hit)?
- b. Si suponemos una tasa de fallos de página de 0,3; que el 70% de las ocasiones la página a reemplazar se encuentra modificada. ¿Cuál será el TAE si el 60% de las veces la entrada de la tabla de páginas se encuentra en la TLB (hit)?
- c. Si suponemos que el 60% de las veces la página a reemplazar está modificada, el 100% de las veces la entrada de la tabla de páginas requerida se encuentra en la TLB (hit) y se espera un TAE menor a 200 nanosegundos.
- d. ¿Cuál es la máxima tasa aceptable de fallas de página?

18. Considere el siguiente programa:

```
#define Size 64
int A[Size; Size], B[Size; Size], C[Size;
Size]; int register i, j;
for (j = 0; j < Size; j++)
    for (i = 0; i < Size; i++)
        C[i; j] = A[i; j] + B[i; j];
```

Si asumimos que el programa se ejecuta en un sistema que utiliza paginación por demanda para administrar la memoria, donde cada página es de 1Kb. Cada número entero (int) ocupa 4 bytes. Es claro que cada matriz requiere de 16 páginas para almacenarse. Por ejemplo:  $A[0,0]..A[0,63]$ ,  $A[1,0]..A[1,63]$ ,  $A[2,0]..A[2,63]$  y  $A[3,0]..A[3,63]$  se almacenará en la primera página.

Asumamos que el sistema utiliza un working set de 4 marcos para este proceso. Uno de los 4 marcos es utilizado por el programa y los otros 3 se utilizan para datos (las matrices). También consideramos que para los índices "i" y "j" se utilizan 2 registros, por lo que no es necesario el acceso a la memoria para estas 2 variables.

- Analizar cuántos fallos de páginas ocurren al ejecutar el programa (considere las veces que se ejecuta  $C[i,j] = A[i,j] + B[i,j]$ )
- Puede ser modificado el programa para minimizar el número de fallos de páginas. En caso de ser posible indicar la cantidad de fallos de páginas que ocurren.

19. Considere las siguientes secuencias de referencias a páginas de los procesos A y B, donde se muestra en instante de tiempo en el que ocurrió cada una (1 a 78):

Proceso A																																						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
1	2	1	3	4	3	3	2	1	1	3	4	5	6	1	2	1	3	3	4	5	6	6	6	5	4	3	1	1	2	3	4	5	4	3	2	1	1	1
4	4	4	4	4	4	4	4	4	4	50	5	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78
0	1	2	3	4	5	6	7	8	9	1	1	2	3	4	5	6	5	1	1	2	3	4	5	4	2	1	1	2	3	4	5	1	1					
2	2	1	2	1	1	2	3	4	5	6	7	8	6	5	3	1	2	3	4	5	6	5	1	1	2	3	4	5	4	2	1	1	2	3	4	5	1	1

Proceso B																																						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
1	2	3	3	3	4	4	5	1	1	1	2	3	4	4	4	4	1	1	2	3	6	5	6	5	4	6	1	1	1	1	4	5	1	3	2	1	1	2
4	4	4	4	4	4	4	4	4	4	50	5	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78
0	1	2	3	4	5	6	7	8	9	1	1	2	3	4	5	6	5	1	1	2	3	4	5	4	2	1	1	2	3	4	5	1	1					
2	2	3	3	4	4	4	3	4	5	4	6	1	1	1	2	3	3	3	4	5	5	5	1	6	2	3	1	2	3	1	1	1	2	2	2	2	3	3

- Considerando una ventana  $\Delta=5$ , indique cuál sería el conjunto de trabajo de los procesos A y B en el instante 24 ( $WS_A(24)$  y  $WS_B(24)$ )
- Considerando una ventana  $\Delta=5$ , indique cuál sería el conjunto de trabajo de los procesos A y B en el instante 60 ( $WS_A(60)$  y  $WS_B(60)$ )
- Para los WS obtenidos en el inciso a), si contamos con 8 frames en el sistema ¿Se puede indicar que estamos ante una situación de trashing? ¿Y si contáramos con 6 frames?
- Considerando únicamente el proceso A, y suponiendo que al mismo se le asignaron inicialmente 4 marcos, donde el de reemplazo de páginas es realizado considerando el algoritmo FIFO. ¿Cuál será la tasa de fallos en el instante 38 de páginas suponiendo que la misma se calcula contando los fallos de páginas que ocurrieron en las últimas 10 unidades de tiempo?
- Para el valor obtenido en el inciso d), si suponemos que el S.O. utiliza como límites superior e inferior de tasa de fallos de páginas los valores 2 y 5 respectivamente ¿Qué acción podría tomar el S.O. respecto a la cantidad de marcos asignados al proceso?

**Administración de E/S**

## 20. Dispositivos

- a. Los dispositivos, según la forma de transferir los datos, se pueden clasificar en 2 tipos:
  - i. Orientados a bloques
  - ii. Orientados a flujos
- b. Describa las diferencias entre ambos tipos.
- c. Cite ejemplos de dispositivos de ambos tipos.
- d. Enuncie las diferencias que existen entre los dispositivos de E/S y que el SO debe considerar.

## 21. Técnicas de E/S. Describa cómo trabajan las siguientes técnicas de E/S

- a. E/S programada
- b. E/S dirigida por interrupciones
- c. DMA (Acceso Directo a Memoria)

## 22. La técnica de E/S programa puede trabajar de dos formas:

- a. E/S mapeada
- b. E/S aislada

Indique cómo trabajan estas 2 técnicas.

## 23. Enuncie las metas que debe perseguir un SO para la administración de la entrada/salida.

## 24. Drivers

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

## 25. Realice un gráfico que marque la relación entre el Subsistema de E/S, los drivers, los controladores de dispositivos y los dispositivos.

## 26. 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.

## 27. 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.

## 28. Enuncie qué servicios provee el SO para la administración de E/S.

**Administración de Discos**

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



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

- *Seek Time*
- *Latency Time*
- *Transfer Time*

De una definición para estos tres tiempos.

31. Suponga un disco con las siguientes características:

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

32. 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 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 (*First Come, First Served*)
- b. SSTF (*Shortest Seek Time First*)
- c. Scan
- d. Look
- e. C-Scan (*Circular Scan*)
- f. C-Look (*Circular Look*)

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

34. Supongamos un Head con movimiento en 300 pistas (numerados de 0 a 299), se encuentra 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
- b. SSTF
- c. Scan
- d. Look
- e. C-Scan
- f. C-Look

35. Supongamos un Head con movimiento en 300 pistas (numerados de 0 a 299), que está 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,  $147^P$ , 81, 257, 94, 133, 212, 175, 130; y después de 30 movimientos se incorporan los requerimientos de las pistas 85,  $15^P$ , 202 y 288; y después de otros 40 movimientos más se incorporan los requerimientos de las pistas 75,  $149^P$ , 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
- b. SSTF
- c. Scan
- d. Look
- e. C-Scan
- f. C-Look

### **Administración de Archivos**

36. Dados las siguientes técnicas de administración de espacio de un archivo:

- Asignación contigua
  - Asignación enlazada
  - Asignación indexada
- a. Describa cómo trabaja cada una.
  - b. Cite ventajas y desventajas de cada una.
  - c. Indique para cada técnica si la misma puede producir fragmentación interna y/o externa.

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

- Tabla de bits
  - Lista Enlazada
  - Agrupamiento
  - Recuento
- a. Describa cómo trabajan estos métodos.

- b. Cite ventajas y desventajas de cada uno.

38. Describa la organización del FileSystem de Unix-System V

39. Gestión de archivos en *UNIX* System V

El sistema de archivos de UNIX utiliza una variante del esquema de asignación indexada para la administración de espacio de los archivos. Cada archivo o directorio está 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 campos para direcciones a los bloques de datos, organizadas de la siguiente manera:

- 10 de direccionamiento directo.
  - 1 de direccionamiento indirecto simple.
  - 1 de direccionamiento indirecto doble.
  - 1 de direccionamiento indirecto triple.
- a. Realice un gráfico que represente la estructura del *I-NODO* y de los bloques de datos. Cada bloque es de 1 Kib(Kibibits).
- b. Si cada dirección para referenciar un bloque es de 32 bits:
- i. ¿Cuántas referencias (direcciones) a bloque pueden contener un bloque de disco?
  - ii. ¿Cuál sería el tamaño máximo de un archivo?

40. Analice las siguientes fórmulas necesarias para localizar un I-NODO en la lista de inodos:

**nro bloque** = ((nro de inodo - 1)/nro. de inodos por bloque) + bloque de comienzo de la lista de inodos.

**Desplazamiento del inodo en el bloque** = ((nro de inodo - 1) módulo (número de inodos por bloque)) \* medida de inodo del disco.

- a. Según la primera 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 un bloque de disco con 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?

41. **Laboratorio opcional de Entrada-Salida**

Se recomienda resolver este laboratorio con el Sistema en modo consola y con la menor cantidad de programas en ejecución posible.

- a. Instale e investigue para que sirve el siguiente programa *hdparm*  
`sudo apt-get install hdparm`
- b. Ejecutar comando:  
`sudo hdparm -l /dev/sda`
  - i. ¿Cuántos cilindros, cabezas y sectores tiene su disco?
  - ii. ¿Qué pasa si ejecuta esto en un disco de estado sólido?
- c. Ahora ejecute el siguiente comando varias veces (al menos 5), de manera tal de poder calcular el promedio de resultados obtenidos  
`sudo hdparm /dev/sda`
- d. Ahora ejecutarlo de la siguiente manera:  
`hdparm -t --direct --offset 500 /dev/sda`  
Investigue la finalidad de los parámetros *direct* y *offset*
- e. Compare los tiempos promedios obtenidos con los parámetros *direct* y *offset* y sin ellos (Recuerde comparar tiempos promedio y no ejecuciones aisladas).
- f. ¿Con qué concepto de la teoría asocia el parámetro *direct*?
- g. Ejecute el siguiente comando:  
`ls /sys/block/`
- h. ¿A qué le parece que corresponde cada entrada del directorio anterior?
- i. Ejecute el siguiente comando (ajuste el dispositivo de disco según su pc):  
`cat /sys/block/sda/queue/scheduler`
- j. Investigue el resultado del comando anterior. ¿Qué quiere decir cada ítem del resultado?, investigue cada uno de ellos y asociarlo con conceptos de la teoría y de esta práctica.
- k. ¿Cuál es la diferencia entre los siguientes conceptos?
  - i. *noop*
  - ii. *deadline*
  - iii. *cfq*
  - iv. *Anticipatory IO*
- l. Como root, ejecute el siguiente comando  
`sudo echo "noop" > /sys/block/sda/queue/scheduler`  
¿Cuál es el efecto de esto?. Ayuda: vuelva a ejecutar  
`cat /sys/block/sda/queue/scheduler`