

SISTEMAS OPERATIVOS

Segunda prueba de evaluación a distancia (PED2)

Nombre y apellidos: Rafael Angel Pérez Toribio.

DNI: 05656369W

Centro: 018003-Ciudad Real-Puertollano

1. Explique razonadamente si las siguientes afirmaciones son verdaderas o falsas:

I) La sobrepaginación aumenta el porcentaje de uso del procesador.

Es falsa. Aunque aumentar el grado de multiprogramación del sistema conduce a una mayor utilización del Procesado, si el grado de multiprogramación supera un determinado valor óptimo, se produce el efecto contrario y la utilización del procesador disminuye, dado que la memoria principal no podrá contener los conjuntos de trabajo de todos los procesos cargados, produciéndose constantemente fallos de página (sobrepaginación) y los procesos se bloquearán.

II) Se denomina buffering de páginas a la estrategia consistente en cargar un cierto número de páginas de un proceso antes de iniciar o continuar su ejecución.

Es verdadera. El buffering de páginas consiste en reservar una serie de marcos libres que contienen páginas, a fin de reducir los reemplazamientos en el caso de que una dirección virtual produzca un fallo de página.

1. Un sistema con memoria virtual mediante demanda de páginas utiliza el algoritmo LRU para la sustitución de páginas. Un proceso genera la siguiente secuencia de referencias a páginas de memoria:

1 3 2 4 1 5 7 4 3 2 8 9 4 5 4 9 1 8 3 2

a) Determinar cuántos fallos de página se producen cuando se dispone de 4 o 5 marcos de página para este proceso.

El algoritmo LRU reemplaza la página que lleva más tiempo sin ser referencia (es decir, menos recientemente usada), utilizando la información del pasado reciente para predecir el futuro próximo.

En el caso de que utilicemos 4 marcos de página se producen 16 fallos de página (tal como se observa en la imagen 2a.1). Si utilizamos 5 marcos de página se producirían 14 fallos de página (tal como se observa en la imagen 2a.2)

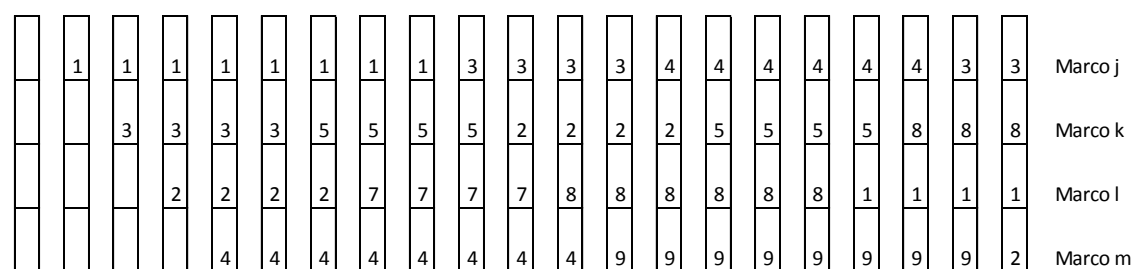
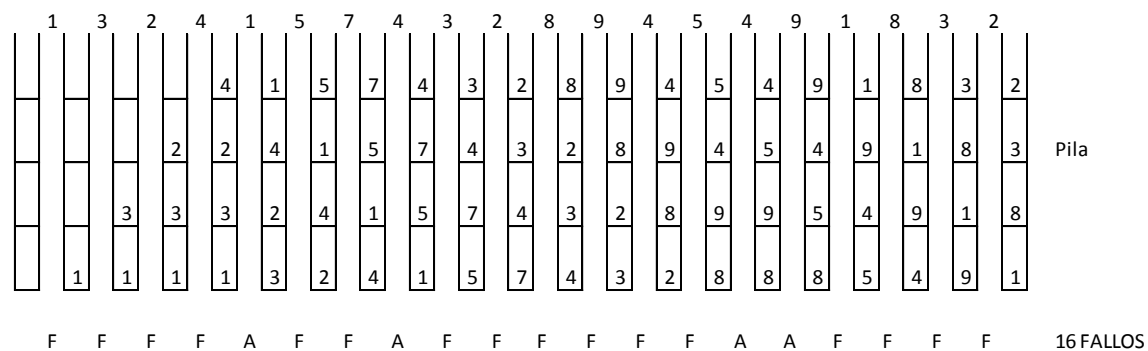


Figura 2a.1

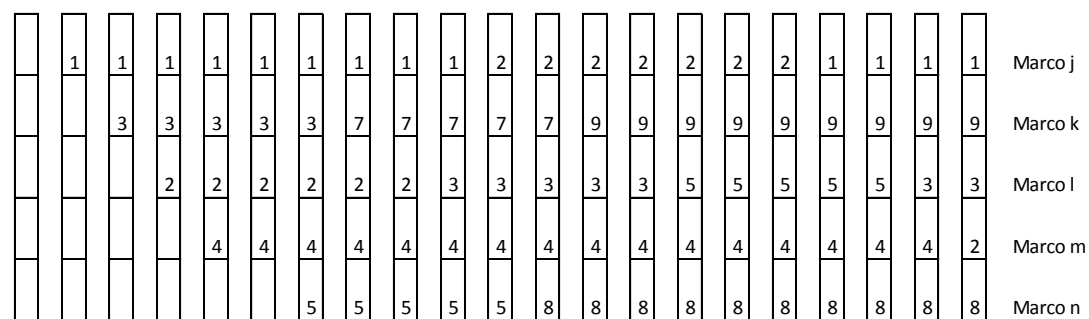
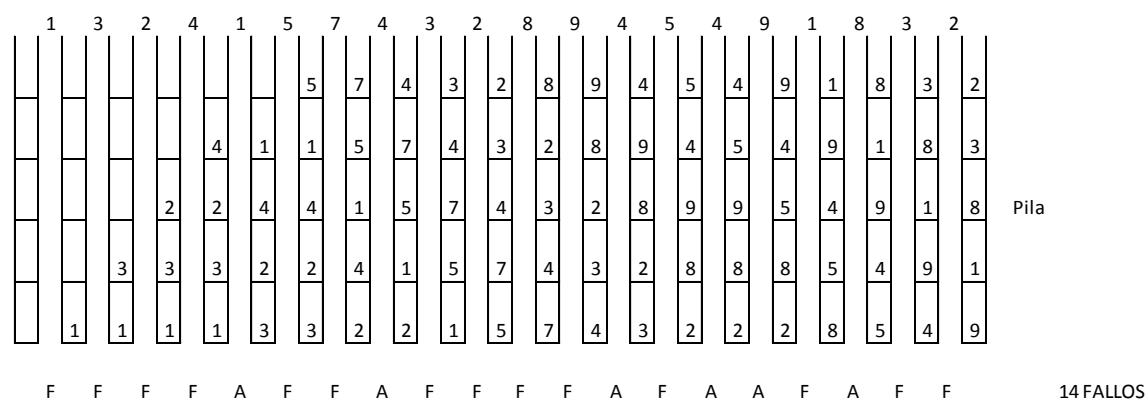


Figura 2a.2

b) Explicar razonadamente si mejoraría la tasa de fallos de página si se aumentase el número de marcos de página a N, siendo N>5

En el caso que nos ocupa, mejoraría esta tasa de fallos. Consideremos:

- Que el número de fallos que se producen es igual a N_p (Siendo N_p el número de páginas diferentes, ya que la primera vez que se referencia, sucede error) + $\sum_{i=1-N_p} (Nf_i)$ (siendo Nf_i el fallo que sucede después de la primera referencia a una página).
- Que la probabilidad de que suceda un fallo de referencia (más allá del suceso seguro de la primera referenciación) es $1-N_m/N_p$ (Siendo N_m el número de marcos de página).

Es decir, si incrementamos el número de marcos, la probabilidad de fallo se irá aproximando a 0 (será 0 cuando $N_m=N_p$), y siempre que exista un número de referencias mayor al número de páginas diferentes, se puede visualizar una mejora en la tasa de fallos.

En el caso que nos ocupa, hay 20 referencias a 9 páginas diferentes, las cuales tendrán como mínimo 9 fallos. Con 4 marcos se alcanzan 16 fallos y con 5 marcos 14 fallos, por lo que la mejora sería posible.

3. Explique razonadamente las funciones que realizan las capas de software de E/S del núcleo de un sistema operativo.

Las capas de software de E/S del núcleo de un sistema operativo son: El subsistema de E/S, los Drivers de dispositivos y los Manejadores de Interrupciones.

El Subsistema de E/S realiza todas las tareas necesarias para las operaciones de E/S independientes de los dispositivos. Sus tareas más importantes son:

- Asignar y liberar los dispositivos dedicados, es decir, aquellos que no pueden ser utilizados simultáneamente por varios procesos.
- Bloquear los procesos que soliciten E/S.
- Planificar la E/S para utilizar de forma eficiente y equitativa los recursos por todos los procesos.
- Invocar o llamar al driver de dispositivo apropiado para la operación de E/S.
- Realiza labores de buffering o almacenamiento temporal que ayuda para la sincronización entre los dispositivos de E/S y los datos solicitados o resultantes de los procesos.
- Proporciona tamaños de bloques lógicos uniformes a los niveles superiores de software.
- Gestiona los errores de las operaciones de E/S, tanto los de programación, es decir aquellos debidos a un fallo de programación o imposibles de realizar, y los de dispositivo, cuando es el propio dispositivo el que ha generado un error o excepción.
- Proporciona una interfaz para los driver de los dispositivos, tanto las que invoca el subsistema de E/S para solicitar un servicio al driver, como las propias del núcleo del sistema operativo que el driver puede invocar.

Los drivers de dispositivos de E/S es el código que permite a un sistema operativo controlar un determinado tipo de dispositivo de E/S (un ratón, un teclado, un disco duro...etc). Existe un driver de dispositivo por cada sistema operativo, facilitado por el fabricante. El driver interactúa con el subsistema de E/S y con el controlador de E/S del dispositivo, suministrando un conjunto de funciones que el Sistema Operativo puede realizar sobre el dispositivo e invocando, del mismo modo funciones del núcleo.

Fundamentalmente realiza las siguientes acciones:

- Comprueba que la función y los parámetros invocados por el subsistema de E/S son correctos para el dispositivo, traduciéndolos a parámetros específicos para dicho dispositivo.
- Comprueba si el dispositivo E/S está ocupado.
- Genera un conjunto de órdenes para el controlador del dispositivo, dependiendo de la petición de E/S.
- Comprueba la ejecución de las órdenes por el controlador, usando mecanismos de sincronización (semáforos, paso de mensajes...etc)
- Comprueba que no se han producido errores en la E/S.
- Examina la cola de peticiones de E/S pendientes.

Los manejadores de interrupciones forman parte del núcleo del sistema operativo y gestionan las interrupciones que generan los controladores de E/S. En general, cada interrupción suministra un número llamado número del vector de interrupciones, que es el índice de una tabla de vectores de interrupción, la cual tiene la dirección de comienzo del manejador de la interrupción. Tienen una alta prioridad de ejecución y su código es pequeño y rápido de ejecutar.

4. En un computador con una capacidad de memoria principal de 64 kibipalabras se utiliza gestión de memoria mediante segmentación. La tabla de segmentos (todos los datos numéricos están en decimal) es la siguiente:

Nº de segmento	Base	Longitud
0	0	7230
1	16384	8191
2	32768	1024
3	8192	356
4	24576	4200

Se pide:

- a) Supuesto que una dirección lógica tiene el mismo tamaño en bits que una dirección física y que consta de los campos [nº de segmento, desplazamiento], determinar el tamaño en bits de cada uno de estos campos.

Dado que existen 5 segmentos, el campo nº de segmento necesitará un tamaño s=3 bits. Para obtener el tamaño mínimo d, supondremos que las longitudes se expresan en palabras y tomaremos el segmento con mayor longitud (segmento 1):

$$\text{Mín } d \{Ss \leq 2^d\}$$

$$\text{Mín } d \{8191 \leq 2^d\}$$

$$d = 13$$

De ese modo:

Nº Segmento 3 bits	Desplazamiento 13 bits
--------------------	------------------------

Longitud total: 16 bits.

b) Determinar a qué direcciones físicas expresadas en decimal corresponden las siguientes direcciones lógicas expresadas en hexadecimal: i) 11AE₁₆ ii) 6190₁₆

i) 11AE (Hex)= 0001000110101110 (en binario)

Nº segmento= 000 (Los tres primeros bits) =0

Nº desplazamiento= 1000110101110= $2^{12}+2^8+2^7+2^5+2^3+2^2+2^1=4526_{10}$

El desplazamiento obtenido es menor que la longitud del segmento 0 (7230) . Por lo que :

Dirección física= base del segmento 0+ desplazamiento =0+4526=4526₁₀.

ii) 6190 (Hex)=0110000110010000 (en binario)

Nº segmento= 011 (tres primeros bits)=3₁₀

Nºdesplazamiento=0000110010000= $2^8+2^7+2^4=400_{10}$

El desplazamiento obtenido es mayor que la longitud del segmento 3 (356), por lo que se producirá un error de direccionamiento por violación del tamaño del segmento.

5. La política de gestión de memoria de un cierto sistema es del tipo demanda de página. El tamaño de una página es de 1 KiB, el tamaño máximo de la memoria virtual es de 4 MiB y el tamaño de la memoria física es de 1 MiB. Se pide:

a) Determinar el tamaño de cada uno de los campos de una dirección virtual y de una dirección física.

Suponemos que la longitud de palabra es un byte.

Comencemos por el tamaño de los campos de una dirección física:

Partiendo del tamaño de la memoria física, C_{mp}, (1 MiB): 2²⁰ bytes. Por lo tanto el mínimo n tal que:

$$\{Cmp \leq 2^n\}.$$

n=20 bits

El tamaño de la página, S_p , es de 1 KiB= 2^{10} bytes. Por lo tanto, el mínimo d (desplazamiento) tal que:

$$\{Sp \leq 2^d\}$$

d=10 bits.

El tamaño f del campo número de marco de página se obtendrá:

$$f=n-d=20-10=10 \text{ bits.}$$

Nº Marco de Página 10 bits	Desplazamiento 10 bits
----------------------------	------------------------

Dirección virtual:

A partir del tamaño máximo de la Memoria Virtual, 4 MiB: 2^{24} bytes. Por lo tanto, el tamaño m en bits de la dirección virtual será; m=24.

El tamaño del campo desplazamiento (d) de la dirección virtual es el mismo que el de la dirección física (d), por lo que, P (tamaño de Nº de página) :

$$P= m-d=24-10=14 \text{ bits.}$$

Nº página 14 bits	Desplazamiento 10 bits
-------------------	------------------------

b) Determinar la capacidad mínima que debe tener la tabla de páginas del proceso de mayor tamaño que se puede ejecutar en el sistema ¿Qué tanto por ciento de la memoria principal ocuparía dicha tabla?

El tamaño de la tabla de páginas coincide con el tamaño del espacio virtual de un proceso. En el caso que nos ocupa debe poder referenciar 2^{14} páginas de dirección virtual. Cada entrada se descompondrá en:

- Número de marco de página: En el caso que nos ocupa serán necesarios 10 bits.
- Validez o presencia: 1 bit
- Protección: 1 bit
- Referenciada: 1 bit
- Modificada: 1 bit

Si la unidad direccionable es una palabra de 1 B, necesitaremos, al menos, 2 palabras: 16 bits, para representar la información de la tabla de contenido:

Así, $2^{14} * 2 = 2^{15}$ B= 32 KiB de capacidad mínima. Un 3,13% de la Memoria Principal.