

Problema 1.- Sea un sistema informático que gestiona la memoria con memoria virtual y paginación. El sistema obedece a las siguientes especificaciones:

- Direccionamiento a nivel de byte.
- Direcciones virtuales de 32 bits, los 20 bits más significativos corresponden al número de página y los 12 menos significativos al desplazamiento dentro de la página.
- Cada entrada de la tabla de páginas ocupa 16 bits repartidos según el siguiente esquema:

Bloqueo Marcos	Validez	Referencia	Protección	Número de marco
1 bit	1 bit	1 bit	3 bits	10 bits

- El tamaño de un entero (int) es de 4 bytes.

A continuación se presenta un fragmento de código de un proceso y la tabla de páginas del proceso en un instante determinado

```
#define MAX 4096
```

```
main()
{
    int i;
    int a[MAX];
    .
    .
    .
    for(i=0;i<MAX;i++)
        a[i]=i;
    .
    .
    .
}
```

Tabla de páginas en hexadecimal y el contador de la tabla de páginas comienza en 0:

0x7ECB
0x7CED
0x7D31
0x15B2
0x7A86
0x7823
0x7B01
0x3FA2
0x5C56
0x54AF

0x7300

Esta tabla es de apoyo para traducir binario-hexadecimal

0	1	2	3	4	5	6	7
0000	0001	0010	0011	0100	0101	0110	0111
8	9	A	B	C	D	E	F
1000	1001	1010	1011	1100	1101	1110	1111

Contesta razonadamente a las siguientes cuestiones:

- (a) Tamaño máximo del espacio de direcciones lógicas del proceso
- (b) Tamaño máximo del conjunto residente del proceso
- (c) En el instante reflejado en la tabla de páginas, ¿Cuántos marcos de memoria física tiene asignado el proceso en este momento?
- (d) Sabiendo que la matriz *a* comienza en la dirección lógica 0x4000, expresado en hexadecimal. ¿Qué porcentaje de la matriz está en la memoria y qué direcciones de memoria física ocupa?
- (e) Suponiendo paginación bajo demanda, ¿Qué elementos de la matriz son los que provocarán fallos de páginas?

Problema 2.- Tenemos un obsoleto disco duro de 32MB que queremos recuperar del desván para un ordenador donde hemos instalado una versión muy antigua de Unix. Este S.O. formatea el disco del siguiente modo:

Sector de arranque	Superbloque	Mapa de bits	Tabla de i-nodos	Bloques de datos
--------------------	-------------	--------------	------------------	------------------

Las especificaciones del sistema son:

- El tamaño de un bloque es 1KB.
- El sector de arranque y el superbloque ocupa 1 bloque cada uno.
- Tamaño de la dirección de un bloque es de 2 bytes.
- El número de i-nodos disponibles es de 1024.
- El i-nodo contiene la siguiente información:
 - Tipo de archivo y protección. (4 bytes).
 - Número de enlaces. (4 bytes).
 - Propietario (4 bytes).
 - Grupo (4 bytes).
 - Tamaño del fichero.(4 bytes).
 - Fecha de creación (4 bytes).
 - Fecha de actualización (4 bytes).
 - Fecha de último acceso (4 bytes).
 - 10 entradas de referencias directas a bloque. (2 bytes*entrada)
 - 4 entradas de referencia indirecta simple a bloque. (2 bytes*entrada)
 - 2 entradas de referencias indirectas de 2º nivel. (2 bytes*entrada).

a) Calcular el tamaño de: (1) mapa de bits, (2) la tabla de i-nodos, (3) máximo efectivo del disco

después de formatear.

- b)** En el caso de no tener limitación de capacidad (disco tan grande como fuera necesario), ¿cuál sería el tamaño máximo teórico de un archivo?
- c)** Un proceso necesita acceder a una posición relativa del fichero “datos.txt” situada en el bloque 530. En el momento de estudio el fichero sólo ha sido abierto y, en consecuencia, actualizada la entrada en la tabla de i-nodos. ¿Cuántos accesos a disco habría que realizar para acceder a la posición indicada?