

## Memoria virtual

### Ejercicios resueltos

---

**Ejercicio 1.** Sea un computador de 20 bits con memoria virtual paginada con páginas de 1 KB y un total de memoria física de 256 KB. Se pide, de forma razonada y breve:

- ¿Cuál es el formato de la dirección virtual? Indique los campos y el número de bits de los mismos.
- ¿Cuál es el número máximo de entradas de la tabla de páginas (de un nivel)?
- ¿Cuántos marcos de página tiene la memoria principal?
- ¿Cuáles son los campos que se incluyen en una entrada de la tabla de páginas? Indique también para qué se utiliza cada uno de los campos.

**Solución:**

- Las páginas ocupan  $1 \text{ KB} = 2^{10}$  bytes. Como la dirección virtual ocupa 20 bits, se emplean  $20 - 10 = 10$  bits para el número de página. Por tanto, el formato emplea los 10 bits superiores de la dirección para representar el número de página y los 10 bits inferiores para representar el desplazamiento dentro de la página.
- El número máximo de entradas de la tabla de páginas coincide con el número máximo de páginas, es decir  $2^{10} = 1024$  entradas.
- El número de marcos de página viene dado por  $256 \text{ KB} / 1 \text{ KB} = 256$  marcos.
- En cada entrada de la tabla de página se incluye, entre otros:
  - Bit de presencia
  - Bit de modificado
  - Bit de validez
  - Bits de permisos
  - Campo en el que se almacena el marco.

**Ejercicio 2.** Considere un computador de 32 bits que dispone de un sistema de memoria virtual que emplea páginas de 16 KB y tiene instalada una memoria principal de 1 GB. Indique de forma razonada:

- El formato de la dirección virtual.
- El número máximo de páginas en este computador.
- El número de marcos de página de este computador.
- El tamaño del bloque que se transfiere entre disco y memoria principal cuando ocurre un fallo de página
- El elemento del computador que genera el fallo de página y quién lo trata.

**Solución:**

a)

Id. página	Desplazamiento
$32 - 14$	$2^{14} = 16 \text{ KB}$
<b>18 bits</b>	<b>14 bits</b>

- $2^{32} / 2^{14} = 2^{32-14} = 2^{18}$  marcos
- $2^{30} / 2^{14} = 2^{30-14} = 2^{16}$  páginas
- El tamaño de una página es **16 KB**
- La **MMU genera la excepción** y **la rutina de fallo de página del sistema operativo trata la excepción**

**Ejercicio 3.** Un computador posee un sistema de memoria virtual implementada mediante paginación que utiliza páginas de 8 KB. El computador proporciona un espacio de memoria virtual de  $2^{32}$  bytes y tiene  $2^{23}$  bytes de memoria física. Si la tabla de páginas correspondiente a un programa en ejecución es la siguiente:

Bit de presencia	Bit de modificado	Marco de página/ Bloque de swap
1	0	1
0	0	7
1	1	9
1	0	14
1	0	8
1	1	3
0	0	25
0	1	16
0	0	23
1	0	78

Se pide:

- Indique el formato de la dirección virtual.
- Indique la dirección física correspondiente a la dirección virtual 0x0000608A.
- ¿Cuál es el tamaño que ocupa el espacio de direcciones virtual de este programa?
- Expresar en MB el tamaño de la memoria principal.

**Solución:**

- El computador tiene páginas de  $8\text{ KB} = 2^{13}$  bytes. Como la memoria virtual es de  $2^{32}$  bytes, se emplean los 19 bits superiores de la dirección para la página y los 13 inferiores para el desplazamiento dentro de la página.
- La dirección 0x0000608A == 0000 0000 0000 0000 0110 0000 1000 1010  
Los 19 bits superiores son 0000 0000 0000 0000 011 = 3. La dirección hace referencia a la página 3, que se encuentra en memoria, en el marco 14 = 1110  
La dirección física es 0000 0000 0000 0001 1100 0000 1000 1010 == 0x 0001C08A
- Este programa ocupa 10 páginas, luego el espacio de direcciones que ocupa es de  $10 \times 8 = 80\text{ KB}$ .
- La memoria principal tiene  $2^{23}$  bytes =  $2^{13}\text{ KB} = 2^3\text{ MB} = 8\text{ MB}$ .

**Ejercicio 4.** Se dispone de un computador (que direcciona la memoria por bytes) con un sistema de memoria virtual que emplea direcciones virtuales de 16 bits y páginas de 2 KB. El computador dispone de una memoria física instalada de 8 KB. Se pide:

- ¿Cuál es el tamaño máximo, en KB, de la memoria virtual que se puede direccionar.
- Indique el número de páginas máximo que puede tener un programa que ejecuta en este computador.
- Indique el formato de la dirección virtual empleada en este computador.
- Indique el tamaño del marco de página.
- Indique el número de marcos de página de la memoria física.
- Indique el formato de la dirección física de este computador.
- ¿Cuál es el número máximo de entradas que puede tener la tabla de páginas asociada a un programa que ejecuta en este computador, asumiendo que se trata de una tabla de páginas de un único nivel?
- Indique al menos dos campos de cada entrada en la tabla de página y diga para que se utilizan.

**Solución:**

- Se pueden direccionar  $2^{16}$  direcciones de memoria. Como el computador direcciona la memoria por byte, en cada dirección se almacena un byte y por tanto el tamaño de la memoria virtual será de 64KB.
- Como cada página es de 2KB, como mucho habrá  $64\text{KB}/2\text{KB} = 32$  páginas.
- El formato de la dirección virtual es el siguiente:

Nº página = 5 bits	Desplazamiento = 11 bits
--------------------	--------------------------

- d) El tamaño del marco de página es igual al tamaño de la página, o sea, 2 KB.
- e) Si la memoria física es de 8KB. Habrá  $8 \text{ KB} / 2 \text{ KB} = 4$  marcos.
- f) El formato de la dirección física es el siguiente:

<b>Nº Marco = 2 bits</b>	<b>Desplazamiento = 11 bits</b>
--------------------------	---------------------------------

- g) Como hay 32 páginas y la tabla de páginas es de un único nivel, tendrá 32 entradas como mucho: una para cada posible página de la memoria virtual.
- h) Posibles campos:  
 P/A = Presente/Ausente, indica si una página se encuentra en Memoria Principal o no.  
 M = Modificado, indica si la página que se encuentra en Memoria Principal ha sido modificada o no, de manera que si se ha modificado cuando se produzca un fallo de página antes de traer una nueva página a Memoria Principal habrá que grabarla en el dispositivo de almacenamiento utilizado para dar soporte a la memoria virtual.  
 Nº de Marco = indica el marco en el que se encuentra una página en caso de estar presente en Memoria Principal

**Ejercicio 5.** Un computador posee un sistema de memoria virtual implementada mediante paginación que utiliza páginas de 4 Kbytes. El computador proporciona un espacio de memoria virtual de  $2^{32}$  bytes y tiene  $2^{18}$  bytes de memoria física. Si la tabla de páginas correspondiente a un programa en ejecución es la siguiente:

Bit de presencia	Bit de modificado	Marcos de página/ Bloque de swap
1	0	1
0	0	8
1	1	9
1	0	14
1	0	5
1	0	7
0	0	25
0	1	16

Se pide:

- a) Indique el formato de la dirección virtual.
- b) Indique la dirección física correspondiente a la dirección virtual 0x00005B83
- c) ¿Cuál es el tamaño que ocupa el espacio de direcciones virtual de este programa?

**Solución:**

- a) Las direcciones virtuales son de 32 bits, se emplean 12 bits para el marco y 20 para el número de página.
- b) Dividiendo la dirección virtual (lógica) entre el tamaño de página, se obtiene el número de página, y el resto de la división es el desplazamiento dentro de la página.

$$23456 / 4096 = 5 \text{ y resto } 2976$$

Si nos fijamos en la tabla de páginas del proceso, la página 5 está en el marco 4, la dirección física se obtiene como:

$$(\text{Nº de marco} \times \text{Tamaño de página}) + \text{Desplazamiento} = (4 \times 4096) + 2976 = 19360$$

- c) Este programa ocupa 8 páginas de 4 KB, es decir ocupa 32 KB.

**Ejercicio 6.** Dado un sistema que emplea un esquema de paginación con las siguientes características:

- El número de páginas por proceso es como máximo de 8096 (8K).
  - El tamaño de página es de 32KB.
  - La memoria física es de 16 MB.
  - El tiempo de acceso a disco es de 120 ms, a memoria de 50 ns y a la TLB de 10 ns.
- a) ¿Cuál es el formato de las direcciones virtuales? Especifica el tamaño de los campos y su significado. Asimismo indica el tamaño del espacio direccionable virtualmente.
- b) ¿Qué es la TLB? ¿Cuál es su función?
- c) Atendiendo al contenido de la TLB y de la tabla de páginas de un proceso cualquiera, indique la dirección de memoria accedida para las siguientes direcciones virtuales (expresadas en hexadecimal):

001A007  
0007100  
00140C2

Téngase en cuenta que las entradas de la tabla de páginas tienen 3 bits de control, de los cuales los dos más significativos se corresponden al bit de presencia y al bit de modificación respectivamente.

2	A1C
5	C04
1	043

0	FF1
1	043
2	A1C
3	203
4	76A
5	C04
6	60F
7	663
8	011

- d) Atendiendo a las páginas accedidas en el apartado anterior, indica si se debe producir escritura a disco en caso de que estas sean sustituidas en memoria. ¿Por qué?
- e) Obtener razonadamente el tiempo de acceso para cada una de las direcciones virtuales del apartado “c”.

**Solución:**

- a) Num. Páginas =  $8096 = 2^{13} \rightarrow 13$  bits  
 Tam. Página = 32 KB =  $2^{15} \rightarrow 15$  bits
- Num. Marcos =  $16 \text{ MB} / 32 \text{ KB} = 2^{24} / 2^{15} = 2^9 \rightarrow 9$  bits
- Significado Campos: Numero de página + Desplazamiento
- Espacio direccionable virtualmente  $\rightarrow 2^{28} = 256 \text{ MB}$
- b) Buffer de traducción adelantada. Es una memoria caché asociativa que guarda los marcos de página donde están almacenadas las páginas más recientemente referenciadas. Su función es la de reducir el tiempo de búsqueda de la dirección real a partir de una dirección virtual, ahorrando el acceso a la tabla de páginas.
- c)  $001A007 \rightarrow 0000\ 0000\ 0001\ 1010\ 0000\ 0000\ 0111$
- Desplaz. = 010 0000 0000 0111  
 Num. Página = 0000 0000 0001 1 (3)
- No en TLB, 203  $\rightarrow 0010\ 0000\ 0011$

Bit Presencia = 0  
Bit Modif. = 0

NO ESTÁ EN MEMORIA PRINCIPAL

0007100 → 0000 0000 0000 0111 0001 0000 0000

Desplaz. = 111 0001 0000 0000  
Num. Página = 0000 0000 0000 0 (0)

No en TLB, FF1 → 1111 1111 0001

Bit Presencia = 1  
Bit Modif. = 1

Num. Marco 1 1111 0001

00140C2 → 0000 0000 0001 0100 0000 1100 0010

Desplaz. = 100 0000 1100 0010  
Num. Página = 0000 0000 0001 0 (2)

En TLB, A1C → 1010 0001 1100

Bit Presencia = 1  
Bit Modif. = 0

Num. Marco 0 0001 1100

d) 001A007 → No está en memoria principal.

0007100 → Bit de modificación 1. Hay que actualizar la página en disco.

00140C2 → Bit de modificación 0. La página no ha sido modificada.

e) 001A007 → TLB (10 ns)+Tabla Páginas(50ns)+Disco(120ms)

0007100 → Tabla Páginas (50ns)+Memoria (50ns)

00140C2 → TLB (10ns)+Memoria (50ns)

**Ejercicio 7.** Considere un computador con las siguientes características:

- CPU de 32 bits, con direcciones virtuales de 36 bits (capacidad para direccionar  $2^{36}$  bytes). Buses de direcciones y datos de 32 bits.
- Memoria virtual paginada, con las siguientes características:
  - Tres niveles de tablas de páginas. Páginas con capacidad de almacenamiento para 8KB
  - Cada entrada de cualquier nivel de tabla ocupa una palabra.
  - Las tablas de páginas de primer y segundo nivel son del mismo tamaño.
  - El tamaño de cada tabla de páginas de tercer nivel es 1 página
  - Tiempo de acceso a TLB de 8ns.
- Memoria principal con tiempo de acceso de 70ns.

Considere que la tabla de aciertos de la TLB es del 90%.

- Describe, para el sistema indicado, los campos en que se puede descomponer una dirección
- Calcule el tiempo medio de acceso a memoria de un dato (para lectura o escritura) considerando únicamente los casos en los que no hay fallo de página

**Solución:**

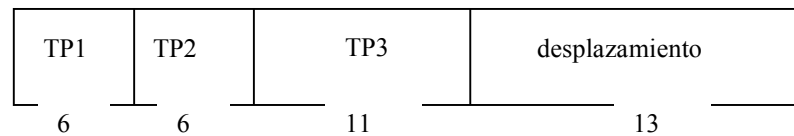


a) Una dirección virtual (36 bits) consta de:

- Desplazamiento dentro de la página
- Número de entrada en la tabla de páginas del 3<sup>er</sup> nivel (TP3)
- Número de entrada en la tabla de páginas del 2<sup>er</sup> nivel (TP2)
- Número de entrada en la tabla de páginas del 1<sup>er</sup> nivel (TP1)

El número de bits empleados en cada uno de los campos es el siguiente:

- Puesto que una página consta de  $2^{13}$  bytes, para determinar el *desplazamiento* dentro de la página se necesitan 13 bits.
- Cada entrada en la tabla de páginas de 3<sup>er</sup> nivel ocupa una palabra, es decir,  $2^2$  bytes y cada TP3 ocupa una página, es decir,  $2^{13}$  bytes por lo que habrá  $2^{13} \div 2^2 = 2^{11}$  entradas. Se necesitan por lo tanto, 11 bits.
- Las tablas de páginas del primer y segundo nivel (TP1 y TP2) son del mismo tamaño, por lo que se utilizará el mismo número de bits para seleccionar una entrada en cada una de ellas. De los 36 bits de que consta una dirección virtual, 24 (13 + 11) se emplean para buscar en TP3 y obtener el desplazamiento dentro de la página, por lo que los 12 restantes se emplearán para determinar las entradas TP1 y TP2. Se utilizarán por lo tanto 6 bits para cada uno de estos campos:



b) Tiempo de acceso promedio a un dato en Memoria Principal:

- Si hay acierto en TLB (probabilidad: 0,9), se completa el acceso en 8ns:  $0,9 \times 8\text{ns} = 7,2\text{ns}$
- Si hay fallo en TLB (probabilidad: 0,1)
  - i. Se accede la Memoria Principal para leer la entrada TP1: 70ns
  - ii. Se accede la Memoria Principal para leer la entrada TP2: 70ns
  - iii. Se accede la Memoria Principal para leer la entrada TP3: 70ns $0,1 \times (70+70+70) = 21\text{ns}$
- Se accede a la memoria Principal para leer (o escribir) el dato: 70ns

$$\text{Tiempo} = 7,2\text{ns} + 21\text{ns} + 70\text{ns} = 98,2\text{ns}$$

**Ejercicio 8.** Sea un computador con las siguientes características:

- Ancho de palabra de 32 bits
- Direcciones de 32 bits
- Memoria física de 64 MB
- Memoria virtual:
  - Tamaño de página: 2 KB
  - Política de reemplazo: FIFO

En un determinado momento, se está ejecutando en este computador un proceso que tiene asignados exclusivamente 8 marcos de página. El fragmento de código correspondiente al periodo que se considera es:

```
s = 0;
m = 0;
r = 0;
for (i = 0; i < 2047; i++)
    s = s + a[i];
for (i = 0; i < 2047; i++)
    m = m + a[i] * b[i];
for (i = 0; i < 2047; i++)
    r = r + a[i] + c[i];
```

La variable de control del bucle ( $i$ ) y las variables  $s$ ,  $m$  y  $r$  están asignadas a registros del procesador. Todas las variables que aparecen son enteros de 32 bits. Las estructuras  $a[i]$ ,  $b[i]$  y  $c[i]$  están asignadas por el procesador a direcciones consecutivas, en ese orden, a partir de la dirección  $H'0000$ .

Considerando sólo los accesos a direcciones de datos, se pide:

- Escriba la traza del programa, indicando la secuencia de páginas a las que se accede.
- Indique los fallos de página que se producen durante la ejecución.
- Repita el apartado b) suponiendo que la política de reemplazo es LRU.

### Solución:

Cada una de las tres estructuras,  $a$ ,  $b$  y  $c$ , ocuparán un total del 2Kpalabras de 4 bytes. Consecuentemente, teniendo en cuenta que el tamaño de página es de 2 Kbytes cada una de ellas ocupará 4 páginas en el sistema de memoria virtual, 12 páginas las tres estructuras. Cada página contendrá 512 componentes consecutivas del vector.

Por otro lado el proceso sólo tiene asignados 8 marcos de página, por lo que necesariamente se producirán fallos de página durante la ejecución del código.

- la estructura  $a$  ocupará la 4 primeras páginas, P0, P1, P2 y P3;  $b$ , las 4 siguientes, P4, P5, P6 y P7; y  $c$ , la restantes, o sea, P8, P9, P10 y P11.

La traza que se solicita sería la sucesión de páginas a las que se accede durante la ejecución:

Primer bucle: acceso solo a los elementos  $a[i]$ :

{P0} [512 veces], {P1} [512 veces], {P2} [512 veces], {P3} [512 veces]

Segundo bucle: acceso alternativo en cada iteración a los elementos  $a[i]$  y  $b[i]$ :

{P0, P4} [512 veces], {P1, P5} [512 veces], {P2, P6} [512 veces], {P3, P7} [512 veces]

Tercer bucle: acceso alternativo en cada iteración a los elementos  $a[i]$  y  $c[i]$ :

{P0, P8} [512 veces], {P1, P9} [512 veces], {P2, P10} [512 veces], {P3, P11} [512 veces]

b) Para determinar los fallos de página que se producen, partimos del supuesto de que ninguna de las páginas está residente en el momento inicial de la ejecución del código.

- Primer bucle: el primer acceso a las cuatro páginas P0, P1, P2 y P3 producirá fallo: 4 fallos
- Segundo bucle: los accesos a P0, P1, P2 y P3 no producirán fallo; sí, los primeros accesos a las páginas correspondientes a  $b$ , P4, P5, P6 y P7: 4 fallos. Con estas 8 páginas ya se habrán completado los marcos disponibles.
- Tercer bucle: el primer acceso a P0 no produce fallo; el primer acceso a cada una de las páginas de la estructura  $c$ , sí: como no hay marcos disponibles se habrá de aplicar la política de reemplazo, FIFO en este caso: las páginas a sustituir serán las correspondientes a la estructura  $a$ , pues son las que llevan más tiempo residentes. En consecuencia, los segundo accesos a P0, P1, P2 y P3, producirán también fallo: 8 fallos

En total, por tanto, con la política de reemplazo FIFO se producen  $(4 + 4 + 8)$  fallos = 16 fallos

c) Si la política de reemplazo que se sigue es LRU (*Least Recently Used*) se reemplazarán siempre las páginas que haya pasado más tiempo sin que se haya accedido a ellas. En nuestro caso particular, sólo se necesitan reemplazos en el tercero de los bucles.

En este caso no se reemplazarían las páginas correspondientes a la estructura  $a$ , pues se vuelven a utilizar en el propio bucle, sino las que corresponden a la estructura  $b$ . Como consecuencia, el número de fallos que se produce es debido sólo a los primeros accesos de  $c$ , mientras que los accesos a páginas de  $a$  encontrarán siempre la página correspondiente

en memoria. Por lo tanto, el número de fallos que se producen es de  $(4 + 4 + 4)$  fallos = 12 fallos, frente a los 16 que ocurrían con la política FIFO.