

ARQUITECTURA DE COMPUTADORES BOLETÍN TEMA 2: JERARQUÍA DE MEMORIA – MEMORIA CACHÉ

Estructura e interpretación

Ejercicio 1. Sea un procesador de 32 bits, con caché de 16KB con líneas de 4 palabras. Dibuje un diagrama con la organización de la caché y la interpretación de las direcciones de memoria principal y caché, e indique en qué posición se ubica la dirección 0xABCDE8F8 en ambas memorias. Suponiendo:

- Caché de mapeado directo.
- Caché asociativa por conjuntos de 4 vías.
- Completamente asociativa.

Ejercicio 2. Se tiene una caché asociativa por conjuntos de 4 vías con 256 líneas por vía. El tamaño de la memoria que almacena las etiquetas es de 2KBytes y el que almacena los datos en la caché es de 8KBytes. Representa la estructura de la caché, así como la interpretación de las direcciones en memoria principal y caché.

Ejercicio 3. Se tiene una caché asociativa por conjuntos de 4 vías, con bloque de 2 palabras de 16 bits, puede contener hasta 4K dobles palabras de memoria y es usado por un procesador de 16 bits con direcciones de 24 bits. Describa la estructura del cache y muestre cómo se interpretan las direcciones. Indique

Ejercicio 4. Un computador con palabras y direcciones de 16 bits tiene una memoria caché de 1K palabras. La memoria caché es asociativa por conjuntos de 4 vías, con un tamaño de bloque de 8 palabras. Especifique la interpretación de las direcciones en memoria principal y caché e indique dónde se ubicaría los bloques correspondientes a las siguientes direcciones 0000110111000011₂, 0x1234, 6234₁₀.

Ejercicio 5. El 80486 tiene una caché interna unificado de 8KBytes, asociativa por conjuntos de 4 vías con bloques de 4 dobles palabras. Posee un total de 128 conjuntos. Hay un bit de línea válida y 3 bits para el algoritmo pseudo-LRU. En caso de fallo de caché, el 486 lee un bloque de 16 bytes de memoria. Dibuje con el máximo detalle la estructura interna de la caché y señale cómo se interpretan las direcciones.

Ejercicio 6. Calcular el tamaño total (incluyendo etiquetas y bits de control) de una caché de CB-WA 64KBytes de datos, bloques 32 bytes y direcciones de 32 bits en los siguientes casos:

- Caché de mapeado directo.
- Caché asociativa por conjuntos de 4 vías.
- Completamente asociativa.

Ejercicio 7. Sea una memoria caché de 128 bytes con política de ubicación asociativa por conjuntos de 4 vías, con 4 conjuntos en total. Las direcciones físicas están compuestas por 32 bits.

- Realice la interpretación de las direcciones físicas de caché y de memoria principal, realizando en ambos casos un dibujo representativo de cada una de las memorias.
- ¿Dónde se ubicará la dirección física 0x000010AF en la caché?
- Si las direcciones físicas 0x000001AF y 0xFFFF7Axy pueden ser asignadas simultáneamente al mismo conjunto caché, ¿qué valores pueden tener x e y?

Accesos sin tipo de fallos

Ejercicio 8. Sea un procesador con una caché de 64 bytes y una memoria principal de 64Kbytes. Suponga que se produce la siguiente secuencia de accesos (algunas direcciones están en decimal y otras en hexadecimal):

	1	2	3	4	5	6	7	8	9	10
Dirección	4	5	0x06	67	68	69	70	0xCF	0xE0	0xE1

Especifique la descomposición de la dirección de la caché, la frecuencia de fallos y cómo queda la memoria tras la secuencia anterior en los siguientes casos:

- Mapeado directo con bloques de 2 bytes.
- Asociativa por conjuntos de 2 vías con bloques de 4 bytes
- Asociativa por conjuntos de 4 vías con bloques de 16 bytes
- Completamente asociativa con bloques de 4 bytes.

Ejercicio 9. Dada la siguiente secuencia de accesos a memoria de 16 bits expresados en hexadecimal y suponiendo una memoria caché de 16 líneas LRU, etiquetar cada referencia de la lista como un acierto o un fallo y mostrar el contenido final de la caché suponiendo:

	1	2	3	4	5	6	7	8	9	10
Dirección	0x10	0x20	0x14	0xE0	0x24	0x2c	0x10	0x114	0X1E0	0X14

- Mapeado Directo con bloques de 16 bytes
- Asociativa por conjuntos de 2 vías y bloques de 4 bytes
- Completamente asociativa con bloques de 32 bytes.

Ejercicio 10. Sea un sistema con una memoria caché capaz de almacenar 4 bloques en total, con tamaño de bloque de 8 bytes y con una memoria principal de 64Kbytes. La caché inicialmente está vacía. Si es necesario reemplazar algún bloque, éste se elige mediante el algoritmo LRU. El procesador emite la siguiente secuencia de direcciones físicas:

	1	2	3	4	5	6
Dirección	0x32B5	0x32B8	0x4A91	0x4220	0xA727	0xC0AA

Analice cómo evoluciona el contenido de la caché, haciendo un recuento de los fallos de línea en cada uno de estos casos:

- Caché totalmente asociativa.
- Caché de mapeado directo.

Rendimiento

Ejercicio 11. Sea el sistema A que posee una caché con un tiempo de acceso de 2ns y una memoria principal con un tiempo de acceso de 20ns que transfiere con velocidad de transferencia 16ns/byte. En cambio, el sistema B posee una caché de 4ns y su memoria principal posee un tiempo de acceso de 40ns pero transfiere a razón de 10ns/byte. ¿Qué tamaño de bloque debería tener ambos sistemas para que el rendimiento de B sea mayor que el de A, si se supone que la frecuencia de fallos permanece constante en ambos sistemas en un 10%? *Solución $T > 6,67 \rightarrow T >= 7$*

Ejercicio 12. Sea un sistema con un ciclo de reloj de 2ns que posee una caché con una penalización por fallo de 20 ciclos de reloj (tanto en lectura como en escritura), una tasa de fallos de 0.05 fallos por instrucción y un tiempo de acceso a la caché de 1 ciclo de reloj (incluido el tiempo para la detección de acierto).

Ejercicio 13. Calcular el tiempo de acceso medio a memoria Suponiendo que al duplicar el tamaño de la caché se consigue reducir la tasa de fallos de 0.05 a 0.03 fallos, pero provoca que el de acceso a la caché pase a ser de 2 ciclos de reloj. ¿se consigue alguna mejora?

Ejercicio 14. Calcular el tiempo medio de acceso a memoria de un computador con una memoria caché y principal con las siguientes características:

- Tiempo de acceso a memoria caché de 4ns
- Tiempo de acceso a memoria principal de 80ns
- Tiempo de transferencia de bloque entre caché y memoria principal de 120ns
- Políticas de escritura: WT-NWA

En este computador se ha observado que la tasa de aciertos a la memoria caché es del 95 % y que cada 100 accesos, 90 son de lectura.

Ejercicio 15. Sea un computador dotado de una memoria cache con las siguientes características:

- Tamaño: 16KB con bloques de 32 bytes (8 palabras)
- Tiempo de acceso: 10ns

Esta memoria está conectada a través de un bus de 32 bits a una memoria principal que tiene un tiempo de acceso de 40ns y es capaz de transferir 8 bytes cada 10ns. Se pide calcular la tasa de aciertos que es necesaria para que el tiempo medio de acceso al sistema de memoria sea de 20ns.

Ejercicio 16. Se dispone de un computador con una memoria caché con un tamaño de 64 KB para guardar instrucciones o datos de los procesos. El tamaño de bloque es de 64 bytes. Los procesos pueden direccionar 8 MB de memoria principal y ésta se direcciona por bytes. Un acceso a memoria principal consume 80ns y a la caché 30ns. La caché es asociativa por conjuntos de 2 vías.

- Haga un dibujo de la estructura de la caché.
- Indique el número de conjuntos que tiene la caché.
- Diga cuánto tiempo tardaríamos en obtener un dato si se produce un fallo en la caché.

Trazas

Ejercicio 17. Un computador contiene una memoria principal de 32 K palabras de 16 bits y una cache de 4Kpalabras divididas en conjuntos de 4 vías de 64 palabras por bloque. Suponga que la cache está vacía. El procesador hace referencia a las palabras 0, 1, 2, ... 4351. Entonces, se repite esta secuencia 9 veces más. La cache es 10 veces más rápida que la memoria principal. Estime la mejora resultante por el uso de la memoria cache. Suponga que se utiliza una política LRU para la sustitución de bloques.

Ejercicio 18. Se ejecuta el siguiente código C en una máquina con un procesador que dispone de una memoria caché de datos de 256 bytes con bloques de 4 palabras (de 32 bits):

```
int i, j, zancada, vector[256]; // int es del tamaño de la palabra
... // i,j y zancada están en registros
// vector comienza en la dir. 0.
for (i=0; i<10000; i++)
    for (j=0; j<256; j=j+zancada)
        c = vector[j]+5;
```

Suponga que sólo se accede a la memoria caché de datos para acceder al vector (que comienza en la dirección 0 de memoria principal), y se supone que los enteros son palabras. Se pide:

- ¿Cuál es el Miss Rate (tasa de fallos) cuando la memoria caché es de correspondencia directa (directamente mapeada) y zancada=132?
- ¿y cuando es 131?
- ¿Cambiaría algo si la memoria caché fuera asociativa por conjuntos de dos vías?
- Suponiendo que la memoria caché de datos es 20 veces más rápida que la memoria principal y que el tiempo de acceso a memoria caché es de t ciclos, calcule el tiempo total de acceso a memoria en los casos A y B. ¿Cuál es el tiempo medio de acceso en ambos casos?

Ejercicio 19. Sea un computador de 32 bits con una memoria caché para datos de 32 KB y línea de 64 bytes. La caché es asociativa por conjuntos de 2 vías. Considere los dos siguientes fragmentos de código:

```
int m[512][512];
sum = 0;
for (i = 0; i < 512; i++)
    for (j = 0; j < 512; j++)
        sum = sum + m[i][j];

int m[512][512];
sum = 0;
for (i = 0; i < 512; i++)
    for (j = 0; j < 512; j++)
        sum = sum + m[j][i];
```

Considere que la variable sum se almacena en un registro. Calcule la tasa de fallos de memoria caché de los dos fragmentos (observación: la matriz se almacena por filas), considerando sólo los accesos a las matrices y el hecho de que éstas se encuentren almacenadas en memoria principal a partir de la dirección 0.

Ejercicio 20. Sea un computador de 32 bits con una memoria caché para datos de 8 KB, bloque de 64 bytes y LRU. Calcule la tasa de fallos global del siguiente código:

```
double a[1024], b[1024], c[1024]; // El tipo double ocupa 8 bytes
for (int i = 0; i < 1024; i++) // el vector 'a' comienza en la posición 0.
    a[i] = b[i] + c[i];
```

- Mapeado directo con CB-WA
- Totalmente asociativa con WT-NWA
- Asociativa por conjuntos de 2 vías con CB-WA
- Asociativa por conjuntos de 2 vías con WT-NWA
- Asociativa por conjuntos de 4 vías con CB-WA

Ejercicio 21. Sea un microcontrolador de 16 bits con una memoria principal de 1 Mbyte con bloques de 16 bytes y memoria caché de datos de 256 bytes, que ejecuta el siguiente código:

```
int i=0; //El tipo int es entero de 2 bytes.
int acumulador=0; //La variable i se guarda en registro.
int v[20]={...}; //Acumulador se almacena a partir de 0x00130
for (i=0; i < 20 ; i++) //El vector v1 comienza en 0x00000
{
    acumulador = v[i] + i;
}
```

Calcule la frecuencia de fallos en los siguientes casos identificando el tipo de fallo que se produce en cada momento:

- Mapeado directo con política CB-WA
- Mapeado directo con política WT-NWA
- Mapeado directo con política WT-NWA suponiendo: `acumulador = acumulador + v[i]`
- Mapeado directo con política CB-WA suponiendo: `acumulador` en la dirección `0x100`

Ejercicio 22. Sea un computador con palabras de 32 bits y direcciones de 16 bits con una memoria caché de datos de mapeado directo de 128 bytes con bloques de 16 bytes. Calcule la frecuencia de fallos y los tipos de fallos que se producen al ejecutar el siguiente código:

```
int a[12], b[12], i, n, acc=0; // El tipo int es de tamaño la palabra
for(n=0; n<2; n++) // i, n y acc están en registros
    for(i=0; i<12; i++) // El vector 'a' comienza en 0x0008
        acc = acc + a[i] + b[i]; // El vector 'b' comienza en 0x00A8
```

Ejercicio 23. Posee el siguiente código de alto nivel escrito en C que invierte el contenido de un determinado vector en otro, ejecutado en un computador con un bus de direcciones de 20 bits y un bus de datos de 32 bits:

```
int a[12], b[12], i;
for(i=0; i<12; i++)
    a[i] = b[12 - i - 1];
```

Contemplando los accesos a datos (vectores 'a' y 'b'), y sabiendo que se encuentran almacenados de forma consecutiva en memoria principal a partir de la posición 0; responda los siguientes apartados:

- Suponiendo un tamaño de bloque de 32 bytes, dibuje el tamaño de la memoria principal, la interpretación de la dirección de acceso y cómo quedarían los datos almacenados en ella:
- Además de lo anterior, posee una memoria caché de 256 bytes de tamaño para datos, mapeado directo, CB-WA. Dibuje la memoria caché y la interpretación de la dirección:
- Rellene la siguiente tabla con los accesos de los distintos elementos de ambos vectores

Vector (a/b)	Elemento (0, 1, ...)	Lectura/Escritura	Línea de MC	A/F en MC	Tipo de Fallo	Bloque en MP

- El tiempo de acceso a MC es de 3ns, a MP es de 30ns y el tiempo de transferencia de 1 byte desde MP a MC es de 1ns. Con el MR del apartado anterior, indique el tiempo de acceso medio a su jerarquía de memoria.

Ejercicio 24. Sea un computador de 32 bits con el juego de instrucciones del MIPS, que ejecuta el siguiente fragmento de código cargado a partir de la dirección 0x00000000

```
addi $5, $0, 1000
xor $1, $1, $1
xor $2, $2, $2
bucle: addi $1, $1, 1
      addi $2, $2, 4
      bne $1, $5, bucle
```

Este computador dispone de una memoria caché asociativa por conjuntos de 4 vías, de 32 Kbytes y bloque de 16 bytes. Calcule de forma razonada el número de fallos de caché y la tasa de aciertos que produce el fragmento de código anterior, asumiendo que se ejecuta sin ninguna interrupción y que la memoria caché está inicialmente vacía. Tenga en cuenta que deberá realizar los cálculos sobre los accesos a instrucciones, ya que no se accede a ningún dato de caché en todo el programa. Suponga, además, que las instrucciones se encuentran en memoria principal a partir de la dirección 0.

Ejercicio 25. Sea un computador MIPS de 32 bits que ejecuta el código expuesto a continuación:

```
addi $3, $0, 3
addi $2, $10, 36
for: lw $5, 4($10)
     add $1, $5, $3
     sw $1, 0($10)
     addi $10, $10, 4
     slt $4, $10, $2
     bne $4, $0, for
     sw $0, 0($10)
```

Suponiendo que \$10 contiene el valor 0x00000000 y que las instrucciones se encuentran de forma consecutiva almacenadas a partir de la dirección 0x10000000:

- Calcule el miss rate de la ejecución de este programa en un sistema de memorias cachés separadas, ambas de mapeado directo de tamaño 64KB, tamaño de bloque de 4 palabras, CB-WA y política de reemplazo random.
- Calcule el miss rate de la ejecución de este programa en un sistema de memoria caché unificada de mapeado directo de tamaño 64KB, tamaño de bloque de 4 palabras, CB-WA y política de reemplazo random.
- Calcule el miss rate de la ejecución de este programa en un sistema de memoria caché unificada asociativa por conjuntos de dos vías de tamaño 64KB, tamaño de bloque de 4 palabras, CB-WA y política de reemplazo random.

Ejercicio 26. Sea un computador MIPS de 32 bits que ejecuta el siguiente código:

```
add $4, $0, $0          # i es inicializado a 0, $4 = 0
bucle: add $5, $4, $1    # $5 = dirección de b[i]
      lw $6, 0($5)      # $6 = b[i]
      add $5, $4, $2    # $5 = dirección de c[i]
      lw $7, 0($5)      # $7 = c[i]
      add $6, $6, $7    # $6 = b[i] + c[i]
      add $5, $4, $10   # $5 = dirección de a[i]
      sw $6, 0($5)      # a[i] = b[i] + c[i]
      addi $4, $4, 4    # i = i + 1
      slti $5, $4, 256  # $5 = 1 si $4 < 256, es decir, i < 64
      bne $5, $0, bucle # ir a bucle si $4 < 256
```

Se considera que \$10, \$1, and \$2 representan las direcciones iniciales de los arrays a, b, y c respectivamente. Dichos registros poseen los valores 0x10000000, 0x20000000 y 0x30000000 respectivamente; y las instrucciones se ubican a partir de la dirección 0x00000000 inclusive.

- Calcule el miss rate de la ejecución de este programa en un sistema de memorias cachés separadas, ambas de mapeado directo de tamaño 128KB, tamaño de bloque de 8 palabras, CB-WA y política de reemplazo random.
- Calcule el miss rate de la ejecución de este programa en un sistema de memoria caché unificada de mapeado directo de tamaño 128KB, tamaño de bloque de 8 palabras, CB-WA y política de reemplazo random.
- Calcule el miss rate de la ejecución de este programa en un sistema de memoria caché unificada asociativa por conjuntos de dos vías de tamaño 128KB, tamaño de bloque de 8 palabras, CB-WA y política de reemplazo random.

Globales

Ejercicio 27. Posee un computador de 32 bits con las siguientes características del sistema de memoria:

- Memoria Principal máxima direccionable de 1MByte, con tamaño de bloque de 16Bytes.
- Memoria Caché de 128Bytes de tamaño para datos, Mapeado Directo, CB-WA.
- Tiempos de acceso: 3ns para MC y 25ns para MP.
- Tiempo de transferencia de MP a MC: 1ns/palabra.

Se ejecuta el siguiente código C en su procesador:

```
int a[7]={...}, b[7]={...}; //Recuerde: un "int" equivale a una palabra.
for(i=0; i<7; i++)
    a[7-i-1] = b[i];
```

Dónde: "a" y "b" se ubican en memoria principal a partir de los bloques 10 y 3, respectivamente.

- Represente la interpretación de la dirección por parte de MC y MP, así como la estructura de la MC.
- Represente la MP, indicando las posiciones de los elementos de cada vector dentro de la misma.
- Rellene una tabla de accesos a los vectores de datos (a y b) y a sus posiciones (0, 1, 2, ...), similar a la mostrada a continuación. Nota: rellene por orden de acceso a memoria.

Vector	Posición	Lect./Escrit.	Dir. física (hex)	Bloque MP	Línea MC	A/F (y tipo) MC

- Calcule el tiempo medio de acceso a su jerarquía de memoria.
- Si la caché fuera WT-NWA, indique cómo variaría el miss rate y en qué casos. Justifique su respuesta y calcule el miss rate resultante.

Ejercicio 28. Sea un computador de 32 bits con una memoria una caché WT-NWA y LRU de 512 bytes de correspondencia directa con un tamaño de línea de 16 bytes. Suponga que se ejecuta el siguiente programa:

```
// El tipo int ocupa 32 bits.
int rep,i,acum=0; //rep e i está en registro, acum está a partir de la dir. 52
int x[14] = {...} // x comienza en la dirección 568
int y[14] = {...} // y comienza en la dirección 1640
for (rep=0; rep<4; rep++)
    for (i=0; i<14; i = i + 2)
        acum = x[i] + y[i]
```

- Especifique mediante un dibujo cómo se interpreta la dirección con la que se accede a la memoria caché y la estructura de ésta con todo detalle (todos los campos y el tamaño de cada uno de ellos)
- Represente la ubicación de las variables del programa en memoria principal especificando claramente los bloques de memoria que ocupa y su posición dentro de éstos.
- Muestre de forma simplificada dónde se cargan los bloques en memoria caché a medida que se ejecuta el programa. Para ello, la caché deberá mostrar, además de los bloques que han quedado cargados tras la ejecución, los bloques que han sido reemplazados tachándolos.

- d. Rellene una tabla similar a la siguiente con la traza de la última iteración del bucle exterior (controlado por rep) accesos a las variables acum, x e y.

Variable	Dir. de bloque	Línea	Acierto ó Tipo de Fallo	Bloque reemplazado (si procede)

- e. Calcule el tiempo medio de acceso a memoria obtenido al ejecutar el programa sabiendo que se necesita 2ns en leer/escribir en caché, 20ns para acceder a memoria principal y 5ns por byte para leer/escribir en memoria principal. Indique los cálculos realizados.
- f. Indique cómo quedaría la caché tras la ejecución y calcule la frecuencia de fallos si la caché fuese CB-WA de 4 vías.

Ejercicio 29. Posee un procesador de 32 bits con las siguientes características del sistema de memoria:

- Memoria Principal máxima direccionable de 64KBytes, con tamaño de bloque de 16Bytes.
- Memoria Caché de 64Bytes de tamaño para datos, Mapeado Directo.
- Tiempos de acceso: 2ns para MC y 20ns para MP.
- Tiempo de transferencia de MP a MC: 1ns/palabra.

Se ejecuta el siguiente código C en su procesador:

```
double a[5]={...}, b[5]={...}; //Recuerde: un "double" equivale a 2 palabras
for(i=0; i<5; i++)
    a[i] = b[4-i];
```

Dónde: "a" se ubica a partir de la dirección 0x48 y "b" a partir de la 0x20 (ambas inclusive).

- a. Represente el sistema al completo.
- b. Rellene una tabla similar a la siguiente, que contenga los accesos a los datos del programa anterior teniendo en cuenta los dos posibles tipos de MC según sus políticas de escritura (CB-WA y WT-NWA). Nota: rellene por orden de acceso a memoria según el código C.

Vector (a/b)	Elemento (0, 1, 2, ...)	Lectura/ Escritura	Dir. física (hex)	Línea MC accedida	A/F (y tipo) de MC si CB-WA	A/F (y tipo) de MC si WT-NWA

- c. Calcule el tiempo medio de acceso a su jerarquía de memoria en ambos casos (CB-WA y WT-NWA).
- d. Represente cómo quedaría rellena la MC, tras finalizar la ejecución del programa anterior, para cada uno de los casos anteriores.

Ejercicio 30. Posee el siguiente código de alto nivel escrito en C que invierte el contenido de un determinado vector en otro, ejecutado en un computador con un bus de direcciones de 20 bits y un bus de datos de 32 bits:

```
int a[12], b[12], i;
for(i=0; i<12; i++)
    a[i] = b[12 - i - 1];
```

Contemplando los accesos a datos (vectores 'a' y 'b'), y sabiendo que se encuentran almacenados de forma consecutiva en memoria principal a partir de la posición 0; responda los siguientes apartados:

- a. Suponiendo un tamaño de bloque de 32 bytes, dibuje el tamaño de la memoria principal, la interpretación de la dirección de acceso y cómo quedarían los datos almacenados en ella:
- b. Además de lo anterior, posee una memoria caché de 256 bytes de tamaño para datos, mapeado directo, CB-WA. Dibuje la memoria caché y la interpretación de la dirección:
- c. Rellene la siguiente tabla con los accesos de los distintos elementos de ambos vectores

Vector (a/b)	Elemento (0, 1, 2...)	Lectura/Escritura	Línea MC	A/F MC	Tipo Fallo	Bloque MP

- d. El tiempo de acceso a MC es de 3ns, a MP es de 30ns y el tiempo de transferencia de 1 byte desde MP a MC es de 1ns. Con el MR anterior, indique el tiempo de acceso medio a su jerarquía de memoria.

Ejercicio 31. Posee un procesador de 16 bits con las siguientes características del sistema de memoria:

- Memoria Principal máxima direccionable de 1KByte, con tamaño de bloque de 8Bytes.
- Memorias Cachés separadas de 32Bytes de tamaño cada una, CB-WA, Mapeado directo.

El siguiente código intercambia el contenido de dos matrices de datos en orden inverso:

Instrucciones (de 32 bits)	Código en C
Instrucción 0	int i, a[6], b[6]; //un "int" equivale a una palabra
Instrucción 1	i = 0;
Instrucción 2	while(i<6) {
Instrucción 3	temp = a[5-i];
Instrucción 4	a[5-i] = b[i];
Instrucción 5	b[i] = temp;
	i++; }

Dónde: "a" se ubica a partir de la dirección 0x40 de MP (inclusive), "b" a partir de la dirección 0x78 de MP (inclusive) y las instrucciones a partir de la dirección 0x10 (inclusive).

- a) Represente gráficamente todo lo que necesite acerca de su sistema de memoria. Deje indicado los cálculos realizados (en caso contrario no se tendrá en cuenta la solución). Incluya el contenido de la MP (en las posiciones correspondientes) contabilizando únicamente los vectores y las instrucciones.
- b) Rellene la tabla de accesos a los datos del programa (si algún campo no procede, indíquelo con "--", no lo deje en blanco). Rellene por orden de acceso a memoria según el código C y solo accesos a vectores.

Vector y Elemento (a[0], b[1],...)	Lectura/Escritura	Dirección Física	Línea MC accedida	A/F y tipo fallo	Bloque MP accedido

- c) Represente cómo se irían actualizando los bloques en la MC de datos durante la ejecución del programa. Para ello, vaya escribiendo el número de bloque de MP que se ubica en cada línea de MC de datos y, a medida que se vayan sustituyendo, vaya tachándolos y colocando a su derecha el nuevo bloque que entra.
- d) Rellene la siguiente tabla de accesos a las instrucciones del programa (si algún campo no procede, indíquelo con "--"). Represente únicamente las dos primeras iteraciones del bucle.

Instrucción	Dirección Física	Línea MC accedida	A/F y tipo fallo	Bloque MP accedido

- e) Represente cómo se irían actualizando los bloques en MC de instrucciones durante la ejecución del programa. Para ello, vaya escribiendo el número de bloque de MP que se ubica en cada línea y, a medida que se vayan sustituyendo, vaya tachándolos y colocando a su derecha el nuevo bloque que entra.
- f) Calcule el Miss Rate global de su programa (datos e instrucciones).
- g) Explique detallada y justificadamente, para cada una de las modificaciones siguientes, qué implicación tendría sobre el programa anterior en cuanto a Miss Rate de cada caché y variación de los diversos Tipos de Fallos en cada caché.
 - a. Política de escritura WT-NWA
 - b. Aumento de asociatividad a 2, manteniendo el tamaño.

Ejercicio 32. Normalmente un mayor grado de asociatividad proporciona una menor tasa de fallos, pero no siempre. Considere una memoria caché de 512 bytes con bloques de 16 bytes, asociativa por conjuntos de 2 vías (con política de sustitución LRU), y escriba una traza de 6 accesos a memoria con direcciones de 32 bits para la cual esta caché tenga el mismo número de fallos que una del mismo tamaño, pero de correspondencia directa. Estime la tasa de fallos (miss-rate) en cada caso. Haga lo mismo con una traza en la que el número de fallos sea estrictamente mayor en la asociativa frente a la de correspondencia directa.

Ejercicio 33. Normalmente un mayor grado de asociatividad proporciona una menor tasa de fallos, pero no siempre. Considere una memoria caché de 512 bytes con bloques de 16 bytes asociativa de grado 2 (con política de sustitución LRU), y escriba una traza de 6 accesos a memoria con direcciones de 32 bits para la cual esta caché tenga igual o más fallos que una del mismo tamaño, pero de correspondencia directa. Estime la tasa de fallos (miss-rate) en cada caso.

Ejercicio 34. Disponemos de un sistema de memoria caché multinivel, cuyo primer nivel está separado en caché de datos e instrucciones, pero con el segundo nivel unificado. El sistema de memoria que posee, junto a su contenido, se detalla a continuación:

MC L1-Datos (CB-NWA)

	V	D	Et.	Bloq.
L0				
L1				
L2				
L3				

MC L1-Instr. (WT-NWA)

	V	Et.	Bloq.
L0			
L1			
L2			
L3			

Mem. Principal
Bloque (8B)

B0		
...		
B11	Ins0	Ins1
B12	Ins2	Ins3
...		
B17	A	B
B18	C	D
...		
B36	E	F
...		
B52	Ins4	Ins5
...		
B1023		

MC L2 (CB-WA, Pseudo-LRU de 1 bit)

	Vía 0					Vía 1					Vía 2				
	V	D	P _{LRU}	Et.	Bloq.	V	D	P _{LRU}	Et.	Bloq.	V	D	P _{LRU}	Et.	Bloq.
C0															
C1															

- Represente la interpretación de las direcciones físicas llevada a cabo por cada MC y por la MP, en las que se indiquen cada uno de los campos que las componen, así como los cálculos realizados para obtenerlo y el tamaño total de cada una de las memorias (teniendo en cuenta todos los campos).
- Con la siguiente secuencia de accesos a memoria (y teniendo en cuenta el contenido inicial que se indicó al comienzo del ejercicio), complete la siguiente tabla y represente, sobre la jerarquía de memoria indicada anteriormente, cómo quedaría el contenido de cada una de las memorias. Indique, además, la frecuencia de fallo de cada una de las memorias caché, así como la frecuencia de fallos global de la MC.

Contenido accedido	Lectura/Escritura	Dirección Física	Línea MC nivel 1	A/F MC nv. 1 (y tipo fallo)	Cjto y Vía MC nivel 2	A/F MC nv. 2 (y tipo fallo)	Bloque MP
Instr. 0	Lectura						
Dato A	Lectura						
Dato F	Escritura						
Instr. 1	Lectura						
Instr. 2	Lectura						
Dato E	Lectura						
Dato E	Escritura						
Instr. 4	Lectura						
Instr. 5	Lectura						
Instr. 0	Lectura						
Dato B	Lectura						
Dato C	Escritura						
Instr. 1	Lectura						
Instr. 2	Lectura						
Dato E	Lectura						
Dato E	Escritura						
Instr. 4	Lectura						
Instr. 5	Lectura						
Instr. 3	Lectura						
Dato D	Escritura						