

Relacion-de-problemas-del-tema-3...



L0R3N



Estructura de Computadores



2º Grado en Ingeniería Informática



**Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga**



Si tu ordenador tiene más años que los
que dura tu carrera: toca cambio.



Katana GF66

¿Necesitas un portátil para estudiar pero
eres de los que se echa una partidita para
desconectar?



ELIGE DESTINO
Y JUÉGALO EN

Marvel's
Spider-Man y cientos
de juegos para viajar
donde quieras.

Consigue esta oferta con
las Tarjetas Regalo
PlayStation en



TE FALTA FNAC | FNAC.ES

Promoción válida en PlayStation Store hasta el 20 de diciembre

NEW
YORK
A UN
SALTO

La ciudad
que nunca
duerme

365 DÍAS POR

89^{,99} €

¡SOLO AHORA!

campus virtual
ETSI Informática

CV ► ETSI Informática ► Mis asignaturas en este Centro ► Curso académico 2019-2020 ► Grado en Ingeniería del Software ►
Estructura de Computadores (2019-20. Grupo C, Gr. Ingeniería del Software, Gr. Ingeniería de Computadores, Gr. Ingeniería Informática) ► Tema 3: Jerarquía de Memoria ►
Relación de problemas del tema 3 (memoria)

Comenzado el	lunes, 16 de diciembre de 2019, 12:36
Estado	Finalizado
Finalizado en	lunes, 16 de diciembre de 2019, 20:10
Tiempo empleado	7 horas 33 minutos
La puntuación	140,00/144,00
Calificación	9,72 de 10,00 (97%)

Pregunta 1

Correcta

Puntuía 6,00 sobre 6,00

Marcar pregunta

Un computador tiene un bus de direcciones de 32 bits y es direccionable a nivel de byte. La cache tiene 256 Kbytes, con palabras de 1 byte y bloques de 32 bits, **asignación directa**, reemplazo LRU y post-escritura. Se utiliza un contador de 5 bits para el reemplazo y 1 bit de Dirty para los bloques que hayan sido modificados y por tanto tienen que escribirse en memoria principal.

a) Descompón en los siguientes campos la dirección física e indica su tamaño en bits:

TAG = 14 bits

Indice/conjunto = 16 bits

w (palabra dentro del bloque) = 2 bits

b (byte dentro de la palabra) = 0 bits

b) ¿Detectas algún dato en el enunciado que no te parezca consistente?

Si, con una asignación directa no hace falta implementar una política de reemplazo (no necesito el contador de 5 bits en el directorio) ▼

¿Cuál es el tamaño, en bits, del directorio de la cache? Considera que en el directorio se encuentran todos los bits de control necesarios para el funcionamiento de la memoria cache.

Tamaño 1048576 bits

Pregunta 2

Correcta

Puntuía 6,00 sobre 6,00

Marcar pregunta

Una memoria cache **asociativa por conjuntos de 2 bloques** con direcciones de 32 bits y palabras de 1 byte tiene 4 bytes en cada uno de sus bloques y una capacidad total de 128 Kbytes. El direccionamiento es a nivel de byte. Se pide:

a) El tamaño de los siguientes campos de la dirección de memoria principal:

- el campo número de línea o bloque: 30 bits
- etiqueta (TAG): 16 bits
- conjunto (c): 14 bits
- desplazamiento de palabra dentro del bloque (w): 2 bits

b) Indicar el valor de cada campo para las siguientes direcciones de memoria principal. Debes proporcionar los valores en hexadecimal, con el número de dígitos que corresponde al tamaño del campo y usando mayúsculas para las letras:

Dirección MP	TAG	c	w
0284A482h	0284	2920	2
01148C89h	0114	2322	1
0038CF00h	0038	33C0	0
0038CF01h	0038	33C0	1

c) ¿Pueden todos los bloques que incluyen las referencias anteriores estar en caché al mismo tiempo? Si ▼

Pregunta 3

Correcta

Puntuía 8,00 sobre 8,00

Marcar pregunta

Sea un sistema de memoria de 1 MByte, con palabras de 1 byte, que incorpora una memoria caché de 64 KBytes **organizada asociativamente en cuatro conjuntos** con 8 palabras por bloque y algoritmo de reemplazo LRU. Se ejecuta un programa que referencia a la secuencia de direcciones de memoria principal (en hexadecimal) que aparece en la tabla.

a) Indica para cada referencia la dirección de bloque en memoria principal (en hexadecimal), el conjunto de la caché asignado y si se produce un acierto o un fallo. Para ello habrás tenido previamente que determinar para tus referencias:

- Tamaño del campo que identifica el byte dentro de la palabra: 0 bits
- Tamaño del campo que identifica la palabra dentro del bloque: 3 bits
- Tamaño del campo que asigna el conjunto: 2 bits
- Tamaño del campo etiqueta (TAG): 15 bits

Dirección MP	Bloque en MP	Conjunto	Acierto
ABC80h	15790	0	Fallo
ABC81h	15790	0	Acierto
ABC88h	15791	1	Fallo

Dirección MP	Bloque en MP	Conjunto	Acierto
BCD90h	179B2 ✓h	2 ✓h	Fallo ▼ ✓
BCD9Dh	179B3 ✓h	3 ✓h	Fallo ▼ ✓
BCDA0h	179B4 ✓h	0 ✓h	Fallo ▼ ✓
CDE00h	19BC0 ✓h	0 ✓h	Fallo ▼ ✓
CDE18h	19BC3 ✓h	3 ✓h	Fallo ▼ ✓
CDE20h	19BC4 ✓h	0 ✓h	Fallo ▼ ✓

Calcula el índice de fallos (con dos decimales): ✓

b) Supón que las referencias anteriores se corresponden con una iteración de un bucle. ¿Cuál será el índice de fallos cuando el número de iteraciones tienda a infinito? ✓

Pregunta 4

Correcta

Puntúa 10,00 sobre 10,00

✓ Marcar pregunta

Disponemos de una memoria cache de 4 Kbytes con tamaño de bloque de 256 bytes, **asociatividad 2** y *algoritmo de reemplazo LRU*. Conectamos la cache entre un procesador que trabaja con palabras de 8 bits y una memoria principal de 1 Mbyte. Para la siguiente secuencia de peticiones de direcciones a memoria principal:

319F0h, 31AF0h, 7013Ch, 77777h, 44037h, 778DEh, A5021h

a) Indica para una referencia:

- Tamaño del campo que identifica el byte dentro de la palabra: bits ✓
- Tamaño del campo que identifica la palabra dentro del bloque: bits ✓
- Tamaño del campo que asigna el conjunto: bits ✓
- Tamaño del campo etiqueta (TAG): bits ✓
- Suponiendo la cache inicialmente vacía, describir la evolución del directorio cache identificando, la etiqueta (TAG, en hexadecimal), el conjunto de la caché asignado (c, en hexadecimal) y si se produce un acierto o un fallo.

Secuencia de direcciones

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acuerdo / Fallo
319F0h	063 ✓h	1 ✓h	Fallo ▼ ✓
31AF0h	063 ✓h	2 ✓h	Fallo ▼ ✓
7013Ch	0E0 ✓h	1 ✓h	Fallo ▼ ✓
77777h	0EE ✓h	7 ✓h	Fallo ▼ ✓
44037h	088 ✓h	0 ✓h	Fallo ▼ ✓
778DEh	0EF ✓h	0 ✓h	Fallo ▼ ✓
A5021h	14A ✓h	0 ✓h	Fallo ▼ ✓

• Calcula el índice de fallos (con dos decimales): ✓

b) Comparar el sistema anterior con respecto a una cache **organizada de forma directa**. En este caso, indica para una referencia:

- Tamaño del campo que identifica el byte dentro de la palabra: bits ✓
- Tamaño del campo que identifica la palabra dentro del bloque: bits ✓
- Tamaño del campo que asigna el conjunto: bits ✓
- Tamaño del campo etiqueta (TAG): bits ✓
- Suponiendo la cache inicialmente vacía, describir la evolución del directorio cache identificando, la etiqueta (TAG, en hexadecimal), el conjunto de la caché asignado (c, en hexadecimal) y si se produce un acierto o un fallo.

Secuencia de direcciones

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acuerdo / Fallo
319F0h	31 ✓h	9 ✓h	Fallo ▼ ✓
31AF0h	31 ✓h	A ✓h	Fallo ▼ ✓
7013Ch	70 ✓h	1 ✓h	Fallo ▼ ✓
77777h	77 ✓h	7 ✓h	Fallo ▼ ✓
44037h	44 ✓h	0 ✓h	Fallo ▼ ✓
778DEh	77 ✓h	8 ✓h	Fallo ▼ ✓
A5021h	A5 ✓h	0 ✓h	Fallo ▼ ✓

• Calcula el índice de fallos (con dos decimales): ✓

c) Comparar el sistema anterior con respecto a una cache **totalmente asociativa**. En este caso, indica para una referencia:

- Tamaño del campo que identifica el byte dentro de la palabra: bits ✓

Ábrete la Cuenta Online de BBVA y llévate 1 año de Wuolah PRO

cómo??



Las ventajas de **WUOLAH PRO**



Di adiós a la publi en los apuntes y en la web



Descarga carpetas completas de un tirón



participa gratis en todos los sorteos

Ventajas Cuenta Online de BBVA

0€

Sin comisión de administración o mantenimiento de cuenta.
(0 % TIN 0 % TAE)

0€

Sin comisión por emisión y mantenimiento de Tarjeta Aqua débito.

0

Sin necesidad de domiciliar nómina o recibos.

- Tamaño del campo que identifica la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits
- Suponiendo la cache inicialmente vacía, describir la evolución del directorio cache identificando, la etiqueta (TAG, en hexadecimal), el conjunto de la caché asignado (c, en hexadecimal) y si se produce un acierto o un fallo.

Secuencia de direcciones

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
319F0h	319h	0h	Fallo
31AF0h	31Ah	0h	Fallo
7013Ch	701h	0h	Fallo
77777h	777h	0h	Fallo
44037h	440h	0h	Fallo
778DEh	778h	0h	Fallo
A5021h	A50h	0h	Fallo

- Calcula el índice de fallos (con dos decimales):

d) A la vista de los resultados, teniendo en cuenta criterios de rendimiento y coste, recomendarías:

Org. directa

Pregunta 5

Correcta

Puntúa 10,00 sobre 10,00

Marcar pregunta

Un adicto a los PCs pretende evaluar el comportamiento de la caché L1 de datos del procesador "Pear P3" para un famoso videojuego de moda. Dispone para ello de una arquitectura con 16 Mbytes de memoria principal direccionable a nivel de byte y con caché L1 para datos 8 Kbytes. En cuanto al videojuego (programa a ejecutar), se compone de un *bucle de 100 iteraciones*, referenciando en cada una de ellas a una secuencia de 4 datos ubicados en las siguientes direcciones de memoria principal (en hexadecimal): 000A40h, 004A40h, 008A40h, 00BA40h. Nuestro amigo observa que la caché de datos evoluciona de la siguiente manera para la primera iteración del juego (en las 99 restantes, el comportamiento es muy similar):

Al final de la primera referencia:			Al final de la segunda referencia:			Al final de la tercera referencia:			Al final de la cuarta referencia:		
C	L	Etiqu	C	L	Etiqu	C	L	Etiqu	C	L	Etiqu
0	0	-	0	0	-	0	0	-	0	0	-
	1	-		1	-		1	-		1	-
1	0	-	1	0	-	1	0	-	1	0	-
	1	-		1	-		1	-		1	-
2	0	-	2	0	-	2	0	-	2	0	-
	1	-		1	-		1	-		1	-
3	0	-	3	0	-	3	0	-	3	0	-
...	...	-	-	-	-
81	1	-	81	1	-	81	1	-	81	1	-
82	0	000h	82	0	000h	82	0	008h	82	0	008h
	1	-		1	004h		1	004h		1	00Bh
83	0	-	83	0	-	83	0	-	83	0	-
	1	-		1	-		1	-		1	-
84	0	-	84	0	-	84	0	-	84	0	-
...	...	-	-	-	-
126	1	-	126	1	-	126	1	-	126	1	-
127	0	-	127	0	-	127	0	-	127	0	-
	1	-		1	-		1	-		1	-

El significado de cada campo es el siguiente: C = Número de conjunto; L = Número de línea; Etiqu = Campo etiqueta del directorio caché; "-" = Línea vacía).

a) En función del comportamiento mostrado en los diagramas anteriores, determina:

- Tamaño conjunto de los campos que identifican el byte dentro de la palabra y la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits
- Qué estrategias de reemplazo de líneas pueden dar lugar al resultado anterior (excluyendo RANDOM):
- Índice de aciertos total (con dos decimales):

b) Nuestro amigo quiere ahora cambiar el procesador de su equipo, dudando entre adquirir un "Carrot C4" o un "Grape G5".

El "Carrot C4" dispone de 16 Kbytes para la caché de datos y las mismas características que en el "Pear P3" original, aumentando únicamente el nivel de asociatividad (esto es, **se tiene el mismo número de conjuntos, pero se dobla el número de bloques por conjunto**). Por lo tanto ahora los tamaños de los campos pasan a ser:

- Tamaño conjunto de los campos que identifican el byte dentro de la palabra y la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits

Muestra la evolución para las 2 primeras iteraciones en Carrot C4:

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
--------------	----------------	--------------	-----------------

WUOLAH

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
000A40h	000 ✓h	52 ✓h	Fallo ▼ ✓
004A40h	004 ✓h	52 ✓h	Fallo ▼ ✓
008A40h	008 ✓h	52 ✓h	Fallo ▼ ✓
00BA40h	00B ✓h	52 ✓h	Fallo ▼ ✓
000A40h	000 ✓h	52 ✓h	Acierto ▼ ✓
004A40h	004 ✓h	52 ✓h	Acierto ▼ ✓
008A40h	008 ✓h	52 ✓h	Acierto ▼ ✓
00BA40h	00B ✓h	52 ✓h	Acierto ▼ ✓

Con lo que el índice de aciertos total (con dos decimales) sería: ✓

Por otro lado, el "Grape G5" tiene una caché de datos de 32 Kbytes, pero mantiene el nivel de asociatividad y el resto de parámetros del "Pear P3", con la salvedad del **número de conjuntos, que es ahora cuatro veces superior**. Por lo tanto ahora los tamaños de los campos pasan a ser:

- Tamaño conjunto de los campos que identifican el byte dentro de la palabra y la palabra dentro del bloque: bits ✓
- Tamaño del campo que asigna el conjunto: bits ✓
- Tamaño del campo etiqueta (TAG): bits ✓

Muestra la evolución para las 2 primeras iteraciones en Grape G5:

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
000A40h	000 ✓h	052 ✓h	Fallo ▼ ✓
004A40h	001 ✓h	052 ✓h	Fallo ▼ ✓
008A40h	002 ✓h	052 ✓h	Fallo ▼ ✓
00BA40h	002 ✓h	1D2 ✓h	Fallo ▼ ✓
000A40h	000 ✓h	052 ✓h	Fallo ▼ ✓
004A40h	001 ✓h	052 ✓h	Fallo ▼ ✓
008A40h	002 ✓h	052 ✓h	Fallo ▼ ✓
00BA40h	002 ✓h	1D2 ✓h	Acierto ▼ ✓

Con lo que el índice de aciertos total (con dos decimales) pasa a ser: ✓

¿Cuál de los dos microprocesadores recomendarías a nuestro amigo atendiendo exclusivamente al índice de aciertos a caché de datos producido por las referencias a memoria efectuadas desde el mencionado bucle del videojuego?:

✓

Pregunta 6

Correcta

Puntúa 12,00 sobre 12,00

✓ Marcar pregunta

Considérese un procesador con instrucciones de 32 bits y una **caché de instrucciones de 32 bytes con bloques de 8 bytes**. Para los dos fragmentos de código MIPS siguientes, indicar la organización (y sus parámetros) que da lugar a una ejecución con el mínimo número de fallos (si hay varias que produzcan el mismo rendimiento, ordenar por coste hardware). NOTA: El número a la izquierda de cada instrucción indica la dirección en memoria principal donde se encuentra.

Código A:

```
0 lw $1, 100($2)
4 add $1, $1, $3
8 sub $4, $5, $1
12 j 24
...
24 mul $2, $2, $8
28 sub $2, $2, $9
32 div $8, $1, $4
36 j 0
```

Código B:

```
0 lw $1, 100($2)
4 add $1, $1, $3
8 sub $4, $5, $1
12 j 32
...
24 mul $2, $2, $8
28 sub $2, $2, $3
32 add $8,
36 j 24
```

a) Indica para una referencia:

- Para la **organización directa**:
 - Tamaño del campo que identifica el byte dentro de la palabra: bits ✓
 - Tamaño del campo que identifica la palabra dentro del bloque: bits ✓
 - Tamaño del campo que asigna el conjunto: bits ✓
 - Tamaño del campo etiqueta (TAG): bits ✓
- Para la **organización totalmente asociativa**:
 - Tamaño del campo que identifica el byte dentro de la palabra: bits ✓
 - Tamaño del campo que identifica la palabra dentro del bloque: bits ✓
 - Tamaño del campo que asigna el conjunto: bits ✓



Tu ordenador lo único que necesita programar es su jubilación.



El Stealth 15M es uno de los portátiles gaming más finos y ligeros. Siempre menos es más. Ve a donde quieras llevando siempre el máximo rendimiento.

- o Tamaño del campo etiqueta (TAG): 29 ✓ bits
- Para la **organización asociativa por conjuntos**:
 - o Tamaño del campo que identifica el byte dentro de la palabra: 2 ✓ bits
 - o Tamaño del campo que identifica la palabra dentro del bloque: 1 ✓ bits
 - o Tamaño del campo que asigna el conjunto: 1 ✓ bits
 - o Tamaño del campo etiqueta (TAG): 28 ✓ bits

b) Suponiendo que el código se ejecuta un número de iteraciones muy grande (infinito), calcula para la ejecución completa:

Código A:

- Para la **organización directa**:
 - o El índice de fallos (con dos decimales): 0,25 ✓
- Para la **organización totalmente asociativa**:
 - o El índice de fallos (con dos decimales): 0 ✓
- Para la **organización asociativa por conjuntos**:
 - o El índice de fallos (con dos decimales): 0 ✓
- A la vista del rendimiento, teniendo en cuenta el coste hardware recomendaría: Org. asociativa por conjuntos ✓

Código B:

- Para la **organización directa**:
 - o El índice de fallos (con dos decimales): 0 ✓
- Para la **organización totalmente asociativa**:
 - o El índice de fallos (con dos decimales): 0 ✓
- Para la **organización asociativa por conjuntos**:
 - o El índice de fallos (con dos decimales): 0 ✓
- A la vista del rendimiento, teniendo en cuenta el coste hardware recomendaría: Org. directa ✓

Pregunta 7

Correcta

Puntuá 12,00 sobre 12,00

Marcar pregunta

Sea el siguiente programa en MIPS:

```
addi $1, $0, 1
j lab1
lab0: lw $1, 16($0)
sw $1, 80($0)
j lab1
...
lab1: sw $0, 16($0)
lw $2, 208($0)
lw $3, 336($0)
sw $2, 400($0)
bne $1, $0, lab0
```

donde la dirección lab0 es igual a 8 d y lab1 es igual a 140 d. Implementamos el primer nivel de la jerarquía de memoria con una cache de instrucciones, con asignación directa, y otra cache de datos, asociativa de 4 conjuntos y reemplazo LRU. Ambas caches tienen bloques de 8 bytes y un tamaño de 128 bytes cada una. Para cada cache se pide:

a) Mostrar la secuencia de referencias (direcciones, a nivel de byte) que hace el programa durante su ejecución:

- a.1) Secuencia de referencias a instrucciones de MP (en decimal): 0 ✓ d, 4 ✓ d, 140 ✓ d, 144 ✓ d, 148 ✓ d, 152 ✓ d, 156 ✓ d, 8 ✓ d, 12 ✓ d, 16 ✓ d, 140 ✓ d, 144 ✓ d, 148 ✓ d, 152 ✓ d, 156 ✓ d
- a.2) Secuencia de referencias a datos de MP (en decimal): 16 ✓ d, 208 ✓ d, 336 ✓ d, 400 ✓ d, 16 ✓ d, 80 ✓ d, 16 ✓ d, 208 ✓ d, 336 ✓ d, 400 ✓ d

b) Muestra la evolución de cada una de las caches, indicando para cada referencia a un bloque de MP el número de conjunto de caché asignado y si se produce un fallo o un acierto.

b.1) Caché de instrucciones:

En primer lugar, indica para una referencia:

- o Tamaño del campo que identifica el byte dentro de la palabra: 2 ✓ bits
- o Tamaño del campo que identifica la palabra dentro del bloque: 1 ✓ bits
- o Tamaño del campo que asigna el conjunto: 4 ✓ bits
- o Tamaño del campo etiqueta (TAG): 25 ✓ bits

Para las referencias que anotaste en el apartado a.1 indica:

Bloque de MP Conjunto Acierto?

Bloque de MP	Conjunto	Acierto?
0	0	Fallo
0	0	Acierto
17	1	Fallo
18	2	Fallo
18	2	Acierto
19	3	Fallo
19	3	Acierto
1	1	Fallo
1	1	Acierto
2	2	Fallo
17	1	Fallo
18	2	Fallo
18	2	Acierto
19	3	Acierto
19	3	Acierto

b.2) Caché de datos:

En primer lugar, indica para una referencia:

- Tamaño del campo que identifica el byte dentro de la palabra: 2 bits
- Tamaño del campo que identifica la palabra dentro del bloque: 1 bits
- Tamaño del campo que asigna el conjunto: 2 bits
- Tamaño del campo etiqueta (TAG): 27 bits

Para las referencias que anotaste en el apartado a.2 indica:

Bloque de MP	Conjunto	Acierto?
2	2	Fallo
26	2	Fallo
42	2	Fallo
50	2	Fallo
2	2	Acierto
10	2	Fallo
2	2	Acierto
26	2	Fallo
42	2	Fallo
50	2	Fallo

c) Calcula las tasas de fallo (deja indicado numerador y denominador):

Tasa de fallos para la caché de instrucciones: 8 / 15

Tasa de fallos para la caché de datos: 8 / 10

Pregunta 8

Parcialmente correcta

Puntúa 4,00 sobre 8,00

Marcar pregunta

Sea el siguiente programa en MIPS:

```
ori $2, $0, 1000h
loop: lw $1, 2800h($2)
      sub $4, $1, $0
      jal rotar
      sw $7, 7800h($2)
      sw $1, C800h($2)
      subi $2, $2, 4
      bne $2, $0, loop
---
rotar: add $10, $4, $4
      muli $7, $10, 2
      jr $31
```

Se desea usar una memoria cache de tamaño 4 Kbytes, con asignación directa, para resolver las referencias a datos. Se pide:

a) Vamos a analizar primero cual es el tamaño de bloque óptimo teniendo en cuenta la tasa de fallos. Prueba distintos tamaños de bloque, empezando con 1 palabra por bloque hasta llegar al máximo número de palabras por bloque, y calcula sus tasas de fallo.

¿Cuál es la mejor tasa de fallos que obtienes? 1

WUOLAH

b) Si ahora fijamos el tamaño de bloque en 256 bytes: ¿es posible reducir la tasa de fallos aumentando el nivel de asociatividad?. Y si fuera así, ¿partir de qué nivel de asociatividad no se obtiene mejora?

Menor número de fallos totales obtenidos 51

c) Para la configuración elegida en el apartado anterior, ¿Qué política de reemplazo es mejor, LRU o FIFO?

Política de reemplazo óptima

Pregunta 9

Correcta

Puntúa 6,00 sobre 6,00

Marcar pregunta

En la figura se proponen dos códigos en alto nivel que inicializan ambos una matriz de 10×10 números reales a cero. El código generado por el compilador almacena la matriz en la memoria en posiciones consecutivas por filas. Las variables i, j son ubicadas en dos registros diferentes del procesador (no están en memoria).

<pre> i=1; while (i <=10) { j=1; while (j <=10) { A[i][j]=0.0; j=j+1; } i=i+1; }</pre>	<pre> i=1; while (i <=10) { j=1; while (j <=10) { A[j][i]=0.0; j=j+1; } i=i+1; }</pre>
Código A	Código B

Se dispone de un sistema de memoria cuya **caché de datos es completamente asociativa** y con reemplazo FIFO. El tamaño de la caché es equivalente a diez números reales en la representación usada por el procesador y permite almacenar dos números por bloque. Ambos códigos realizan la misma función, pero desde el punto de vista del sistema de memoria el rendimiento es diferente. Analiza cuál de ellos daría más fallos de caché. Para ello contesta a las siguientes preguntas:

- CÓDIGO A:

- En el código A, la secuencia de elementos referenciados es:

A(1,1),A(1,2),A(1,3),A(1,4),...,A(1,10),A(2,1),A(2,2),A(2,3),...,A(10,10)

por lo tanto el acceso a los elementos de la matriz se realiza por Filas

El índice de fallos (expresado como un porcentaje) resultante será: IF= 50 %

- CÓDIGO B:

- En el código B, la secuencia de elementos referenciados es:

A(1,1),A(2,1),A(3,1),A(4,1),...,A(1,2),A(2,2),A(3,2),A(4,2),...,A(10,10)

por lo tanto el acceso a los elementos de la matriz se realiza por Columnas

El índice de fallos (expresado como un porcentaje) resultante será: IF= 100 %

Pregunta 10

Correcta

Puntúa 12,00 sobre 12,00

Marcar pregunta

Queremos ejecutar un bucle simple como éste:

```

for (i=0; i<=9; i++)
  A[i] = 0;
```

en un procesador con una memoria caché de datos de 8 bytes y **tamaño de bloque 2 bytes**. Sabiendo que A es un array de 30 números almacenados en memoria a partir de la dirección 000h y que cada elemento del array ocupa 1 byte, muestra la evolución en la caché de datos de ese código para las siguientes organizaciones:

- Organización directa

Dato	Dirección MP	Dir. Bloque	Conjunto (c)	Acierto / Fallo
A[0]	000h	0	0	Fallo
A[1]	001	0	0	Acierto
A[2]	002	1	1	Fallo
A[3]	003	1	1	Acierto
A[4]	004	2	2	Fallo
A[5]	005	2	2	Acierto
A[6]	006	3	3	Fallo
A[7]	007	3	3	Acierto
A[8]	008	4	0	Fallo
A[9]	009	4	0	Acierto

- Calcula el índice de aciertos (con dos decimales): 0,50

- Organización totalmente asociativa

Dato	Dirección MP	Dir. Bloque	Conjunto (c)	Acierto / Fallo
A[0]	000h	0	0	Fallo

Dato	Dirección MP	Dir. Bloque	Conjunto (c)	Acierto / Fallo
A[1]	001 ✓h	0 ✓h	0 ✓h	Acierto ▼ ✓
A[2]	002 ✓h	1 ✓h	0 ✓h	Fallo ▼ ✓
A[3]	003 ✓h	1 ✓h	0 ✓h	Acierto ▼ ✓
A[4]	004 ✓h	2 ✓h	0 ✓h	Fallo ▼ ✓
A[5]	005 ✓h	2 ✓h	0 ✓h	Acierto ▼ ✓
A[6]	006 ✓h	3 ✓h	0 ✓h	Fallo ▼ ✓
A[7]	007 ✓h	3 ✓h	0 ✓h	Acierto ▼ ✓
A[8]	008 ✓h	4 ✓h	0 ✓h	Fallo ▼ ✓
A[9]	009 ✓h	4 ✓h	0 ✓h	Acierto ▼ ✓

◦ Calcula el índice de aciertos (con dos decimales): ✓

• Organización asociativa por conjuntos

Dato	Dirección MP	Dir. Bloque	Conjunto (c)	Acierto / Fallo
A[0]	000h	0 ✓h	0 ✓h	Fallo ▼ ✓
A[1]	001 ✓h	0 ✓h	0 ✓h	Acierto ▼ ✓
A[2]	002 ✓h	1 ✓h	1 ✓h	Fallo ▼ ✓
A[3]	003 ✓h	1 ✓h	1 ✓h	Acierto ▼ ✓
A[4]	004 ✓h	2 ✓h	0 ✓h	Fallo ▼ ✓
A[5]	005 ✓h	2 ✓h	0 ✓h	Acierto ▼ ✓
A[6]	006 ✓h	3 ✓h	1 ✓h	Fallo ▼ ✓
A[7]	007 ✓h	3 ✓h	1 ✓h	Acierto ▼ ✓
A[8]	008 ✓h	4 ✓h	0 ✓h	Fallo ▼ ✓
A[9]	009 ✓h	4 ✓h	0 ✓h	Acierto ▼ ✓

◦ Calcula el índice de aciertos (con dos decimales): ✓

a) ¿Qué tipo de organización de memoria cache (directa, por conjuntos o totalmente asociativa) elegirías en función del porcentaje de aciertos? Nota: a igualdad de rendimiento se debe elegir la menos costosa desde el punto de vista HW: ✓

b) ¿Se podría decir lo mismo independientemente de la posición de comienzo del array A en memoria?:

✓

Imaginemos ahora que el bucle que estamos procesando es este otro (en la sentencia $A[i] = B[9-i]$ el acceso de lectura ocurre antes que el de escritura):

```
for (i=0; i<=9; i++)
    A[i] = B[9-i];
```

donde A es el mismo array anterior y B es otro array de 45 números de 1 byte, almacenado a partir de la posición 60 de memoria. Realiza un esquema de la evolución en la caché de datos de este código, similar al código anterior, y responde a las siguientes preguntas:

• Organización directa:

◦ Calcula el índice de aciertos (con dos decimales): ✓

• Organización totalmente asociativa:

◦ Calcula el índice de aciertos (con dos decimales): ✓

• Organización asociativa por conjuntos:

◦ Calcula el índice de aciertos (con dos decimales): ✓

c) ¿Qué tipo de organización de memoria cache elegirías ahora?: ✓

d) ¿Es independiente la respuesta anterior de la posición de comienzo del array B en memoria?

✓

Pregunta 11

Correcta

Puntúa 10,00 sobre 10,00

✓ Marcar pregunta

Sea un sistema de memoria de 1 MByte, con palabras de 1 byte, que incorpora una memoria caché de 64 KBytes, con 4096 palabras por bloque y algoritmo de reemplazo FIFO. Se ejecuta un programa que referencia a la siguiente secuencia de direcciones (en hexadecimal):

00000h, 01000h, 01001h, 0F000h, 10000h, 00000h, 40000h, 20000h, 08000h

a) Compara las tasas de fallo cuando se considera una organización con asignación directa, totalmente asociativa y asociativa con 4 conjuntos.

• Organización directa

Dato	Etiqueta (TAG)	Dir. Bloque	Conjunto (c)	Acierto / Fallo
00000h	0 ✓h	00 ✓h	0 ✓h	Fallo ▼ ✓
01000h	0 ✓h	01 ✓h	1 ✓h	Fallo ▼ ✓
01001h	0 ✓h	01 ✓h	1 ✓h	Acierto ▼ ✓

WUOLAH



Un ordenador con el que podrás jugar como mi ex jugó conmigo.



Para que el futuro nos sea más benevolente debemos empezar hoy a perseguirlo, creando para nosotros un camino que pueda llevarnos hacia el lugar donde queramos estar el día de mañana, aprovecha la serie Pulse GL76 y viaja al futuro.

Dato	Etiqueta (TAG)	Dir. Bloque	Conjunto (c)	Acierto / Fallo
0F000h	0 ✓h	0F ✓h	F ✓h	Fallo ▼ ✓
10000h	1 ✓h	10 ✓h	0 ✓h	Fallo ▼ ✓
00000h	0 ✓h	00 ✓h	0 ✓h	Fallo ▼ ✓
40000h	4 ✓h	40 ✓h	0 ✓h	Fallo ▼ ✓
20000h	2 ✓h	20 ✓h	0 ✓h	Fallo ▼ ✓
08000h	0 ✓h	08 ✓h	8 ✓h	Fallo ▼ ✓

Calcula el índice de aciertos (con dos decimales): 0,11 ✓

• Organización totalmente asociativa

Dato	Etiqueta (TAG)	Dir. Bloque	Vía dentro del conjunto	Acierto / Fallo
00000h	00 ✓h	00 ✓h	0 ✓h	Fallo ▼ ✓
01000h	01 ✓h	01 ✓h	1 ✓h	Fallo ▼ ✓
01001h	01 ✓h	01 ✓h	1 ✓h	Acuerdo ▼ ✓
0F000h	0F ✓h	0F ✓h	2 ✓h	Fallo ▼ ✓
10000h	10 ✓h	10 ✓h	3 ✓h	Fallo ▼ ✓
00000h	00 ✓h	00 ✓h	0 ✓h	Acuerdo ▼ ✓
40000h	40 ✓h	40 ✓h	4 ✓h	Fallo ▼ ✓
20000h	20 ✓h	20 ✓h	5 ✓h	Fallo ▼ ✓
08000h	08 ✓h	08 ✓h	6 ✓h	Fallo ▼ ✓

Calcula el índice de aciertos (con dos decimales): 0,22 ✓

• Organización asociativa por conjuntos

Dato	Etiqueta (TAG)	Dir. Bloque	Conjunto (c)	Acierto / Fallo
00000h	00 ✓h	00 ✓h	0 ✓h	Fallo ▼ ✓
01000h	00 ✓h	01 ✓h	1 ✓h	Fallo ▼ ✓
01001h	00 ✓h	01 ✓h	1 ✓h	Acuerdo ▼ ✓
0F000h	03 ✓h	0F ✓h	3 ✓h	Fallo ▼ ✓
10000h	04 ✓h	10 ✓h	0 ✓h	Fallo ▼ ✓
00000h	00 ✓h	00 ✓h	0 ✓h	Acuerdo ▼ ✓
40000h	10 ✓h	40 ✓h	0 ✓h	Fallo ▼ ✓
20000h	08 ✓h	20 ✓h	0 ✓h	Fallo ▼ ✓
08000h	02 ✓h	08 ✓h	0 ✓h	Fallo ▼ ✓

Calcula el índice de aciertos (con dos decimales): 0,22 ✓

b) Calcula el tamaño de cada una de las caches suponiendo que en la zona de directorio solo se almacena la etiqueta TAG.

• Organización directa:

Tamaño de la cache en bits: 524352 ✓ bits

• Organización totalmente asociativa:

Tamaño de la cache en bits: 524416 ✓ bits

• Organización asociativa por conjuntos:

Tamaño de la cache en bits: 524384 ✓ bits

Pregunta 12

Correcta

Puntúa 10,00 sobre 10,00

Marcar pregunta

Considerar una memoria principal de 64 Kbytes direccionable a nivel de byte y con palabras de este mismo tamaño. Sea la siguiente secuencia de direcciones de acceso a memoria principal:

04F5h, 11E0h, 1500h, 2000h, 241Fh, 16FFh, 1233h, 21F0h

Mostrar la evolución, así como los fallos que se producen para cada una de las caches descritas en la siguiente tabla:

	Tamaño Cache	Tamaño Bloque	Organización	Reemplazo
C1	4 Kb	256 bytes	Directa	---
C2	2 Kb	256 bytes	Totalmente asociativa	FIFO
C3	2 Kb	256 bytes	Asociativa por conjuntos de 4 vías	LRU

Cache C1

- Tamaño del campo que identifica el byte dentro de la palabra: 0 ✓ bits

- Tamaño del campo que identifica la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
04F5h	<input type="text" value="0"/> h	<input type="text" value="4"/> h	Fallo ▼
11E0h	<input type="text" value="1"/> h	<input type="text" value="1"/> h	Fallo ▼
1500h	<input type="text" value="1"/> h	<input type="text" value="5"/> h	Fallo ▼
2000h	<input type="text" value="2"/> h	<input type="text" value="0"/> h	Fallo ▼
241Fh	<input type="text" value="2"/> h	<input type="text" value="4"/> h	Fallo ▼
16FFh	<input type="text" value="1"/> h	<input type="text" value="6"/> h	Fallo ▼
1233h	<input type="text" value="1"/> h	<input type="text" value="2"/> h	Fallo ▼
21F0h	<input type="text" value="2"/> h	<input type="text" value="1"/> h	Fallo ▼

- Calcula el índice de fallos (con dos decimales):

Cache C2

- Tamaño del campo que identifica el byte dentro de la palabra: bits
- Tamaño del campo que identifica la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
04F5h	<input type="text" value="04"/> h	<input type="text" value="0"/> h	Fallo ▼
11E0h	<input type="text" value="11"/> h	<input type="text" value="0"/> h	Fallo ▼
1500h	<input type="text" value="15"/> h	<input type="text" value="0"/> h	Fallo ▼
2000h	<input type="text" value="20"/> h	<input type="text" value="0"/> h	Fallo ▼
241Fh	<input type="text" value="24"/> h	<input type="text" value="0"/> h	Fallo ▼
16FFh	<input type="text" value="16"/> h	<input type="text" value="0"/> h	Fallo ▼
1233h	<input type="text" value="12"/> h	<input type="text" value="0"/> h	Fallo ▼
21F0h	<input type="text" value="21"/> h	<input type="text" value="0"/> h	Fallo ▼

- Calcula el índice de fallos (con dos decimales):

Cache C3

- Tamaño del campo que identifica el byte dentro de la palabra: bits
- Tamaño del campo que identifica la palabra dentro del bloque: bits
- Tamaño del campo que asigna el conjunto: bits
- Tamaño del campo etiqueta (TAG): bits

Dirección MP	Etiqueta (TAG)	Conjunto (c)	Acierto / Fallo
04F5h	<input type="text" value="02"/> h	<input type="text" value="0"/> h	Fallo ▼
11E0h	<input type="text" value="08"/> h	<input type="text" value="1"/> h	Fallo ▼
1500h	<input type="text" value="0A"/> h	<input type="text" value="1"/> h	Fallo ▼
2000h	<input type="text" value="10"/> h	<input type="text" value="0"/> h	Fallo ▼
241Fh	<input type="text" value="12"/> h	<input type="text" value="0"/> h	Fallo ▼
16FFh	<input type="text" value="0B"/> h	<input type="text" value="0"/> h	Fallo ▼
1233h	<input type="text" value="09"/> h	<input type="text" value="0"/> h	Fallo ▼
21F0h	<input type="text" value="10"/> h	<input type="text" value="1"/> h	Fallo ▼

- Calcula el índice de fallos (con dos decimales):

Información

Marcar pregunta

**** Solo enunciado ****

En aplicaciones multimedia para la reproducción de audio o vídeo es habitual tener cargas de trabajo denominadas de *streaming*, es decir, cargas de trabajo con una gran cantidad de datos que apenas son reusados. Los datos de audio o de vídeo suelen ser *streams* con datos de tamaño igual a 1 byte que codifican por ejemplo un nivel de gris o de cuantificación de una onda sonora. Considera un *stream* de vídeo que accede a 512 KB de datos secuencialmente:

0, 2, 4, 6, 8, 10, 12, 14, 16...

- a) Considera una cache de 64 Kb con asignación directa y bloques de tamaño 32 bytes. ¿Cuál es la tasa de fallo? ¿Depende del tamaño de la cache o del tamaño de los datos a los que se accede? Usando el modelo de las 3C, ¿qué tipo de fallos se producen?
- b) Recalcula las tasas de fallo considerando bloques de tamaño 16 bytes, 64 bytes y 128 bytes. ¿Qué tipo de localidad se explota en este tipo de trabajo?

La técnica de prefetching (precarga) permite mejorar el acceso a datos cuando el patrón de acceso es predecible. Esta técnica se basa en traer, de forma especulativa, un bloque al que todavía no se ha accedido; por ejemplo, una implementación sencilla consistiría en traer, a un buffer separado, el siguiente bloque adyacente al bloque que se está accediendo. Si el dato que se busca se encuentra en ese buffer se considera un acierto, se introduce en la cache y se precarga el siguiente bloque en el buffer.

c) Considerando que se dispone de un buffer de precarga con capacidad para dos bloques y que la latencia de la cache es menor o igual que el tiempo que necesita nuestro programa para procesar los datos de un bloque de la cache. ¿Cuál sería la tasa de fallos para el *stream* anterior?

Información

Marcar pregunta

**** Solo enunciado ****

El siguiente código en alto nivel:

```
for (i=0; i<2; i++)
  for (j=0; j<2; j++)
  {
    ac = C[i+j*2];
    for (k=0; k<2; k++) ac += A[i+k*2]*B[k+j*2];
    C[i+j*2] = ac;
  }
```

se ejecuta en un procesador que dispone de una memoria principal de 1GB con tamaño de palabra de 4 bytes y con tamaño de línea de 8 bytes. El array C se almacena a partir de la dirección 0 de memoria principal, el array A a partir de la dirección 20 (0x14) y el array B a partir de la dirección 40 (0x28). Se sabe también que cada elemento del array ocupa 4 bytes en memoria. Se pide lo siguiente:

a) Mostrad los accesos a los elementos de los array, junto con su dirección de memoria principal y su número de bloque. Para ello, rellenad una tabla como la de abajo indicando en la columna "dato" el elemento concreto del array que se accede (p.ej, A[3]), la dirección de memoria principal donde se encuentra dicho elemento (dir MP) y número de bloque de memoria principal (bloq MP). Mostradlo en el mismo orden que se solicita en el programa teniendo en cuenta que en cada iteración del tercer bucle FOR, el orden de acceso a los arrays es: B[k+j*2], A[i+k*2].

b) Suponiendo que disponemos de una cache de datos, asociativa por conjuntos de 2 vías (2 bloques por conjunto), reemplazo LRU y tamaño 32 bytes, muestra la evolución del directorio de la cache para dicho código. Para ello, rellena en la tabla la columna "CONJ" con el número del conjunto de la caché donde dicho bloque va a ser alojado, e indica en "A/F" si se produce acierto (A) o fallo (F) de caché y un comentario opcional de lo que ocurre. Muestra también cómo evoluciona el directorio de la cache. Finalmente, calcula el Índice de Fallos (expresálo en forma fraccionaria)

dato	dir MP	bloq MP	CONJ	A/F

Información

Marcar pregunta

**** Solo enunciado ****

El siguiente código en alto nivel:

```
FOR (J=1; J<=2; J++) A[J]=J;
FOR (J=1; J<=3; J++) B[J+1]=2*J;
FOR (J=2; J<=4; J++) A[J+1]=A[J-1]-B[J];
```

se ejecuta en un procesador que dispone de una memoria principal de 1GB con tamaño de palabra de 1 byte y con tamaño de bloque de 2 bytes. El array A se almacena a partir de la dirección 0 de memoria principal y el array B a partir de la dirección 32 (ambos números están expresados en DECIMAL). Se sabe también que cada elemento del array ocupa 1 byte en memoria.

Se pide lo siguiente:

a) Mostrar los accesos a los elementos de los array, junto con su dirección de memoria principal (MP) y su número de línea (bloque). Mostrad los resultados en una tabla similar a la del ejercicio anterior, mostrando el elemento concreto del array que se accede, la dirección de memoria principal donde se encuentra dicho elemento y el número de bloque al que pertenece. Mostradlo en el mismo orden que se solicita en el programa teniendo en cuenta que en cada iteración del tercer bucle FOR, el orden de acceso a los arrays es: B[J], A[J-1], A[J+1].

b) Suponiendo que disponemos de una cache de datos, asociativa por conjuntos de 2 vías (2 bloques por conjunto), reemplazo LRU y tamaño 8 bytes, muestra la evolución del directorio de la cache para dicho código. Para ello, incluye en la tabla anterior dos columnas adicionales: una con el número del conjunto de la caché donde dicho bloque va a ser alojado y otra si se produce acierto (A) o fallo (F) de caché. Complétalo con un comentario adicional de lo que va ocurriendo en cada referencia. Muestra también el estado final en el que queda el directorio de la cache. Finalmente, calcula el Índice de Fallos.

Pregunta 13

Correcta
Puntúa 5,00 sobre 5,00

Marcar pregunta

Supongamos un sistema de memoria compuesto por 2 bancos de memoria principal entrelazada donde el tiempo de direccionamiento es de 1 ciclo de reloj y un tiempo de transferencia de memoria (tiempo de bus) de 2 ciclos. El tiempo de acceso a los bancos de memoria DRAM es de 20 ciclos para el primer acceso y 8 ciclos para los siguientes. Calcula la penalización por fallo y el ancho de banda de la memoria para los siguientes casos:

a) Tamaño de bus: 8 bytes y tamaño de bloque: 32 bytes

Penalización por fallo:  ciclos

WUOLAH

Ancho de banda de la memoria: ✓ bytes/ciclo (redondea a 2 decimales)

b) Tamaño de bus: 8 bytes y tamaño de bloque: 64 bytes

Penalización por fallo: ✓ ciclos

Ancho de banda de la memoria: ✓ bytes/ciclo (redondea a 2 decimales)

Pregunta 14

Correcta

Puntúa 2,00 sobre 2,00

✓ Marcar pregunta

Calcula el tiempo medio de acceso a memoria (AMAT) en nseg. de un sistema cache donde la frecuencia de reloj es de 500 Mhz, el **tiempo de acierto es de 1 ciclo** de reloj, la **penalización por fallo** de 20 ciclos, y donde el **porcentaje de aciertos** es del 95%.

- $t_{AMAT} =$ ✓ ciclos
- $t_{AMAT} =$ ✓ nseg.

Pregunta 15

Correcta

Puntúa 6,00 sobre 6,00

✓ Marcar pregunta

Considera un benchmark que ejecuta 1000 instrucciones en un procesador a 1.2 GHz, del cual 330 instrucciones son load/store. En este programa, el CPI para una cache perfecta es 1.8 (cache perfecta: no hay ningún fallo). Ahora consideremos una cache real con política write-through, en la que la tasa de fallos para la cache de instrucciones (I\$) es del 2% y la de la cache de datos (D\$) es del 8%. La penalización por fallo es de 10 ciclos para I\$ y de 15 ciclos para D\$. Calcula (redondea a 2 decimales los resultados):

- a) Los ciclos de detención (stalls) debidos a los fallos: ✓ ciclos
- b) CPI efectivo: ✓
- c) Tiempo de ejecución del programa: ✓ nseg
- d) Tiempo medio de acceso a memoria (AMAT) (hit time: 1 cc): ✓ nseg

Pregunta 16

Correcta

Puntúa 4,00 sobre 4,00

✓ Marcar pregunta

Considera un procesador con política de write-through cuya I\$ tiene una **tasa de fallos** del 0.5%, mientras que para la D\$ es del 1.5%, la **penalización por fallo** de I\$ es de 10 ciclos y de 15 para la D\$. El CPI base (el de una cache perfecta) es 1.7 y en el código hay un 32% de load&stores. Calcula (redondea a dos decimales si hace falta):

- CPI efectivo= ✓ ciclos
- $t_{AMAT} =$ ✓ ciclos

Pregunta 17

Correcta

Puntúa 8,00 sobre 8,00

✓ Marcar pregunta

Considera un programa que ejecuta 2150 instrucciones en un MIPS a 1.5 GHz en que se han implementado todos los cortocircuitos posibles. Sabemos que el 20% de las instrucciones son saltos condicionales (el 30% de los mismos se toman), el 5% son saltos incondicionales, el 15% son una instrucción de carga seguida de una instrucción con una dependencia de datos verdadera con ella, el 10% son loads sin esa dependencia y el 8% son stores (con política write-through). Adicionalmente la tasa de aciertos para I\$ es del 99,4% (con 12 cc de penalización por fallo) y para la D\$ del 98% (con una penalización por fallo de 15 cc.

El procesador implementa la técnica de predicción estática de salto no tomado, actualizándose el PC en la etapa MEM para los saltos condicionales y en la etapa ID para los incondicionales. **Calcula el CPI resultante.**

Para ello empieza por calcular las penalizaciones (ciclos por instrucción) a añadir sobre el CPI ideal de 1. Redondea a dos decimales donde sea necesario:

- Penalizaciones debidas a riesgos:
 - Penalización causada por los saltos condicionales: ✓
 - Penalización causada por los saltos incondicionales: ✓
 - Penalización debida a los loads-uso: ✓
- Penalizaciones debidas a fallos en acceso a cache:
 - Penalización en las referencias a instrucciones: ✓
 - Penalización en las referencias a datos: ✓

Con lo que al sumarle todas las penalizaciones, el CPI efectivo resultante es: CPIef= ✓

Pregunta 18

Correcta

Puntúa 5,00 sobre 5,00

✓ Marcar pregunta

Considera un sistema cache con dos niveles L1, L2. La tasa de fallos para L1 es del 3% y para L2 es del 5%. Calcula (redondea a dos decimales si te hace falta):

- a) Tasa de fallos global del sistema de memoria: ✓ %
- b) Si el número total de referencias es de 12.000, calcula el número de:
 - i) Fallos en L1: ✓
 - ii) Referencias a L2: ✓
 - iii) Fallos en L2: ✓

c) Asumiendo, como usualmente, que el **"hit time"** para L1 es de 1 cc, el **"hit time"** para L2 es de 10 cc y la **penalización por fallos** de L2 es de 50 cc, calcula el AMAT.



Si tu ordenador tiene más años que los que dura tu carrera: toca cambio.

t_AMAT= ✓ ciclos



¿Necesitas un portátil para estudiar pero eres de los que se echa una partidita para desconectar? Con el Katana GF66 de MSI lo tienes todo; movilidad, autonomía y rendimiento óptimo para compaginar tus dos facetas

Pregunta 19

Correcta

Puntúa 4,00 sobre 4,00

✓ Marcar pregunta

Considera un procesador que trabaja a 1,7 GHz y tiene un sistema cache de dos niveles para el que la tasa de aciertos para L1 es del 90% y para L2 es del 75%. Teniendo en cuenta que el AMAT es de 5 cc y el tiempo de acceso para L2 es de 15 cc, calcula (redondeando a 2 decimales donde sea necesario):

- la penalización por fallo de L1: ✓ ciclos, que equivale a ✓ ns
- la penalización por fallo de L2: ✓ ciclos, que equivale a ✓ ns

Finalizar revisión



uni>ersia



Universidad de Málaga · Avda. Cervantes, 2. 29071 MÁLAGA · Tel. 952131000 · info@uma.es

© Todos los derechos reservados

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH