

soluciones_prueba_Tema3_2018.pdf



Cinder



Arquitectura de Computadores



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



Estamos de
Aniversario

De la universidad al
mercado laboral:
especialízate con los posgrados
de EOI y marca la diferencia.



EOI Escuela de
organización
industrial



saber más

ARQUITECTURA DE COMPUTADORES. Benchmark del Tema 3

Estudiante:

1. En un microprocesador SMT (multihebra simultánea) se pueden enviar a ejecutar varias instrucciones de una misma hebra en un instante determinado. (V)

2. En el protocolo MESI para mantener la coherencia de cache, una línea dada de memoria puede estar, en un momento dado, en el estado E (exclusivo) en una cache y en el estado S (compartido) en otras caches (F)

3. En el protocolo MESI, si en la cache de un nodo N1 hay un bloque B en estado E (Exclusivo), y ese nodo detecta que otro procesador en el nodo N2 intenta leer un dato que está en el bloque B, dicho bloque pasa al estado S (Compartido) en las caches del N1 y N2 (V)

4. En el protocolo MSI, si en la cache de un nodo N1 hay un bloque B en estado M (Modificado), y ese nodo detecta que otro procesador en el nodo N2 intenta escribir un dato que está en el bloque B, dicho bloque pasa al estado I (no válido) en la cache del N1 y a M (Modificado) en N2 (V)

5. En un multiprocesador NUMA con 8 nodos, 16 GBytes por nodo, y líneas de cache de 64 Bytes. ¿Cuántas entradas tiene el directorio de memoria utilizado en cada nodo para mantener la coherencia de cache en un protocolo MSI sin difusión? $16 \text{ GBytes} = 2^{34} \text{ Bytes}$. $\frac{2^{34}}{2^6} = 2^{28} \text{ líneas}$

6. En el multiprocesador NUMA descrito en la pregunta anterior ¿Cuántos bits tienen cada una de las entradas del directorio que se utiliza para mantener la coherencia de cache en un protocolo MSI con directorio? $8+1 = 9 \text{ bits}$

7. En el mismo multiprocesador NUMA anterior con el mismo protocolo de coherencia de cache, se puede tener el estado 1 1 0 ... 0 V (1: hay copia del bloque en la cache del nodo correspondiente al bit; 0: no hay copia en la cache del nodo correspondiente al bit; V: bloque válido en memoria principal) en alguna de las entradas de alguno de los directorios (V)

8. Si el modelo de consistencia de memoria de un multiprocesador NO respeta el orden R→W (Sí respeta todos los demás), e inicialmente X=Y=Z=r1=0 (donde X,Y,y Z son variables en memoria compartida y r1 es un registro de P1), al final se podría tener Y=0 (F)

No puede haber tanto R(X) como W(Y) por el RAW

P1:	P2:
(1) while (Z==0) { };	(a) X=1; W(X)
(2) r1=X; RAW en P1	(b) Y=2; W(Y)
(3) Y=r1;	(c) Z=1; W(Z)

9. Cuando se utilizan instrucciones del tipo LL/SC (lectura enlazada/escritura condicional) para implementar un cerrojo, los recursos hardware asociados a dicha técnica permiten detectar si, entre la ejecución de la lectura enlazada (LL) y la ejecución de la escritura condicional (SC) a la dirección de memoria del cerrojo, algún otro procesador ha accedido a dicha dirección (V)

10. Si en la secuencia de instrucciones siguiente se tiene que r1=1, r2=0, r3=0, dicha secuencia implementa un cerrojo (lock(k)) en el que k=1 significa que el cerrojo está cerrado y k=0 que está abierto. (F)

```

b=r1;
do
    compare&swap(r2,b,k); // si r2==k, k y b se intercambian
while (b==r3);
  
```

ARQUITECTURA DE COMPUTADORES. Benchmark del Tema 3

Estudiante:

1. En un microprocesador SMT (multihebra simultánea), se procesan varias hebras concurrentemente y en un instante determinado solo se pueden enviar a ejecutar instrucciones de una misma hebra.

(F)
2. En el protocolo MESI para mantener la coherencia de cache, una línea puede estar en el estado E (exclusivo) solo en una cache del multiprocesador

(V)
3. En el protocolo MESI, si en la cache de un nodo N1 hay un bloque B en estado S (Compartido), y ese nodo detecta que el procesador del nodo N2 intenta leer un dato que está en el bloque B, dicho bloque pasa al estado S (Compartido) en las caches del N1 y N2

(V)
4. En el protocolo MSI, si en la cache de un nodo N1 hay un bloque B en estado M (Modificado), y ese nodo detecta que otro procesador en el nodo N2 intenta leer un dato que está en el bloque B, dicho bloque pasa al estado I (no válido) en la cache del N1 y a M (Modificado) en N2

(F)
5. En un multiprocesador NUMA con 8 nodos, 8 GBytes por nodo, y líneas de cache de 128 Bytes. ¿Cuántas entradas tiene el directorio de memoria utilizado en cada nodo para mantener la cache en un protocolo MSI sin difusión?

$8\text{ GBytes} = 2^{33}\text{ Bytes.}$
 $\frac{2^{33}}{2^7} = 2^{26}\text{ entradas}$
6. En el multiprocesador NUMA descrito en la pregunta anterior ¿Cuántos bits tiene cada una de las entradas del directorio que se utiliza para mantener la coherencia de cache en un protocolo MSI con directorio?

$8+1=9$
7. En el mismo multiprocesador NUMA anterior, con protocolo MSI de coherencia de chache, es posible que alguna de las entradas de alguno de los directorios esté en el estado 1 1 0 ... 0 I (1: hay copia del bloque en la cache del nodo correspondiente al bit; 0: no hay copia en la cache del nodo correspondiente al bit; I: bloque no válido en memoria principal)

(F)
8. Si el modelo de consistencia de memoria de un multiprocesador NO respeta el orden R→W (Sí respeta todos los demás), e inicialmente X=Y=Z=r1=0 (donde X, Y, Z son variables en memoria compartida y r1 es un registro de P1), al final SOLO se podría tener Y=1

(V)

No puede por el RAW

P1:	P2:
(1) while (Z==0) { ;	(a) X=1; W(X)
(2) r1=X; RAW en r1	(b) Y=2; W(Y)
(3) Y=r1;	(c) Z=1; W(Z)

9. Cuando se utilizan instrucciones del tipo LL/SC (lectura enlazada/escritura condicional) para implementar un cerrojo, los recursos hardware asociados a dicha técnica impiden que, entre la ejecución de la lectura (LL) y la ejecución de la escritura (SC) a la dirección de memoria del cerrojo, ningún otro procesador pueda acceder a dicha dirección de memoria del cerrojo.

(F)
10. Si en la secuencia de instrucciones siguiente se tiene que r1=1, r2=0, r3=1, dicha secuencia implementa un cerrojo (lock(k)) en el que k=1 significa que el cerrojo está cerrado y k=0 que está abierto.

(V)

```
b=r1;
do
    compare&swap(r2,b,k); // si r2==k k y b se intercambian
while (b==r3);
```