Ingeniería de Computadores
Julio 2013
Nombre:

Normas de realización:
- Incluir el nombre en todas las hojas utilizadas
- Todas las respuestas han de ser correctamente detalladas y razonadas.
- Las respuestas deben estar escritas con bolígrafo negro o azul

Pregunta 1 (1 pto). Explica la utilidad de la ventana de instrucciones y explica brevemente cómo puede ser el orden de emisión y el alineamiento en la ventana.

Pregunta 2 (1 pto). Explica brevemente las estructuras que permiten la gestión de predicción de los saltos dinámica explícita

Pregunta 3 (1 pto.). Describe ejemplos de problemas, cuya solución consista en un conjunto de procesos o hebras que se comunican mediante cada uno de los siguientes patrones:

a) Difusión (*broadcast*).
b) Dispersión (*scatter*)

(sólo un ejemplo de problema para cada apartado)
Debe quedar clara la diferencia entre los dos patrones.

Pregunta 4 (1 pto). ¿Cuál es la característica distintiva de las redes de interconexión directas frente a los otros tipos de redes de interconexión (indirectas y de medio compartido)?

Pregunta 5 (1 pto). Indica el nombre de dos topologías de interconexión que cumplan las siguientes características, y dibuja el grafo de cada una de ellas:

- Directa
- Grado 4
- Regular

(las dos topologías deben satisfacer las tres características)

Pregunta 6 (1 pto). Explica brevemente qué tipos de paralelismo existen y en qué niveles puede aplicarse

Pregunta 7 (1 pto). ¿Qué diferencia existe entre el protocolo de coherencia de caché MESI y MSI? ¿Qué ventajas tiene uno de ellos sobre el otro?

Pregunta 8 (1,5 ptos). Dada la siguiente secuencia de instrucciones:

```
lw r1, dato1 ; r1 = dato1
add r1,r1,r0 ; r1 = r1 + r0
lw r2, dato2 ; r2 = dato2
lw r3, dato3 ; r3 =dato3
add r4, r2, r3; r4= r2 + r3
mult r1, r1, r4; r1 = r1*r4
sub r2, r3, r1; r2 = r3 – r1
```

Indica cómo sería la evolución del buffer de reorden y de la tabla de registros. La arquitectura sobre la que se ejecutan las instrucciones está formada por las siguientes etapas: búsqueda de instrucción, decodificación/emisión, ejecución, reorden y escritura. El coste de cada etapa para todas las instrucciones es de 1 ciclo excepto en la etapa de ejecución. En ejecución hay una unidad de carga/almacenamiento, una unidad para suma/resta y una unidad de multiplicación; los ciclos de ejecución según el tipo de instrucción son los siguientes:

- Carga/almacenamiento: 2 ciclos
- Suma/resta: 1 ciclo
- Multiplicación: 3 ciclos

La arquitectura puede captar y decodificar tres instrucciones en paralelo. También pueden finalizar tres instrucciones en paralelo. La estación de reserva es única para todas las unidades de ejecución y admite tres instrucciones como máximo y sigue una política de emisión alineada.
Los valores iniciales de los registros son 0. Los datos son: datoA = 2, datoB = 3, datoC = 4

Pregunta 9 (1,5 ptos). Dos grupos de ingenieros están trabajando en la paralelización de una determinada aplicación. Ambos grupos resuelven que el 20% de la aplicación no es paralelizable. Sin embargo, el primer grupo decide que para el resto, el 25% es paralelizable para cualquier número de nodos y el 75% sólo hasta mitad del número máximo de nodos disponible. El segundo grupo de ingenieros decide, que para la parte paralelizable, se puede paralelizar el 30% para cualquier número de nodos y el 70% sólo hasta un tercio del número de nodos máximo.
Suponiendo que la aplicación va a correr en un multicomputador con N nodos iguales, y que la sobrecarga de comunicación es del el 15% del tiempo de ejecución en paralelo en cada caso, decida cuál es la mejor solución en función de la eficiencia (argumentando así que se trata de la máquina que mejor compromiso tiene entre número de nodos y ganancia en velocidad).

1.

Explica la utilidad de la ventana de instrucciones y explica brevemente cómo puede ser el orden de emisión y el alineamiento de la ventana

La ventana de instrucciones almacena las instrucciones pendientes. Las instrucciones se cargan en la ventana una vez decodificadas y se utiliza un bit para indicar si un operando está disponible o no. Una instrucción puede ser emitida cuando tiene todos sus operandos disponibles y la unidad funcional donde se procesará.

La emisión de una orden puede ser ordenada (sólo lanza las instrucciones en el mismo orden que entran, esperando hasta que la orden esté preparada) o desordenada (lanza las órdenes que estén preparadas, sin esperar a una en particular).

La emisión de las órdenes puede ser alineada (hasta que no se vacía la ventana de instrucciones no se reciben nuevas órdenes) o no alineada (se pueden recibir nuevas órdenes siempre que haya sitio).

2.

Explica brevemente las estructuras que permiten la gestión de predicción de los saltos dinámica explícita

Branch Target Buffer (BTB)
Bits acoplados. Almacena la dirección de los últimos saltos tomados y los bits de predicción de ese salto. Los campos se actualizan después de ejecutar el salto, cuando se conoce si el salto fue tomado o no y la dirección del salto. La predicción implícita no tiene bits de predicción. Lo malo es que sólo se pueden predecir saltos que están en la BTB.

Tabla histórica de datos (BHT)
Bits desacoplados. Tiene 2 tablas:
BTAC, que almacena la dirección de los últimos saltos tomados
BHT, que almacena los bits de predicción de todas las instrucciones de salto condicional
Lo bueno, puede predecir instrucciones que no están en la BTAC, lo malo es que necesitas más hardware

Bits de predicción en la caché
Cuando se capta una instrucción de la caché, si se trata de una instrucción de salto condicional, accede en paralelo a los bits de predicción y si el salto se predice como tomado se accede a la instrucción destino del salto. Para acceder la instrucción destino del salto se utiliza una BTB independiente a la que se añade el índice sucesor a la l-cache. Lo bueno de esta estructura es que también permite predecir saltos que no están en la BTB, pero añadiendo menos hardware que la BHT.

3.

Describe ejemplos de problemas, cuya solución consista en un conjunto de procesos o hebras que se comunican mediante cada uno de los siguientes patrones:
a) Difusión (broadcast)
b) Dispersión (scatter)
(sólo un ejemplo de problema para cada apartado)
Debe quedar clara la diferencia entre los dos patrones.

Respecto a Broadcast,la máquina maestro manda el **mismo mensaje** a las máquinas hijo, por ejemplo al mandar la petición de conexión, o cuando manda una constante con la que deben trabajar.
Por Scatter el maestro divide un mensaje o contenido en **partes diferentes** que luego reparte entre los hijos, como una operación de muchas sumas que divides entre los hijos para que cada uno haga un fragmento de la ecuación.

4.

¿Cuál es la característica distintiva de las redes de interconexión (indirectas y de medio compartido)?

Redes indirectas
Son redes dinámicas (los enlaces entre los nodos de la red son reconfigurables) basadas en conmutadores y árbitros. Tienen topologías regulares (Crossbar, MIN(unidireccionales y bidireccionales) y red de Clos) y topologías irregulares (NOWs).

Redes de medio compartido
Poseen un medio de transmisión compartido. Utilizan un Bus de sistema (arquitectura UMA: Proc - Mem), utilizan un arbitraje para la resolución de conflictos en el bus. Permiten un Sencillo Broadcast, pero tienen un ancho de banda limitado (escalabilidad limitada, cuello de botella).

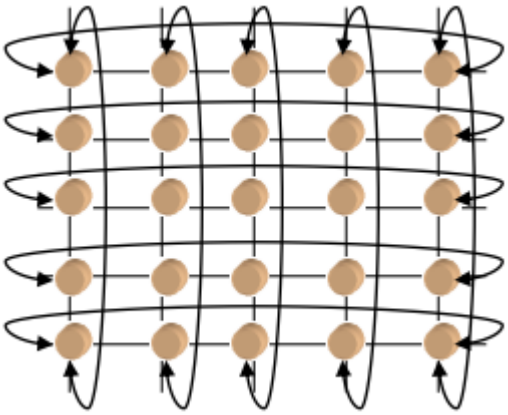
5.

Indica el nombre de dos topologías de interconexión que cumplan las siguientes características, y dibuja el grafo de cada una de ellas

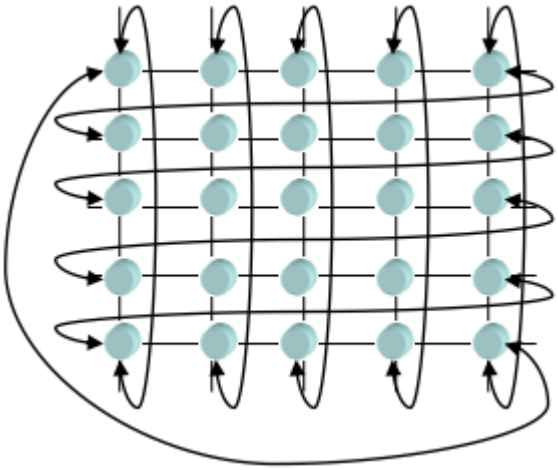
- Directa
- Grado 4
- Regular

(las dos topologías deben satisfacer las tres características)

Red Toro



Malla Illiac



6.

Explica brevemente qué tipos de paralelismo existen y en qué niveles puede aplicarse

Paralelismo de datos
La misma función, instrucción, etc. se ejecuta en paralelo, pero en cada una de esas ejecuciones se aplica sobre un conjunto de datos distinto.

Paralelismo funcional
Varias funciones, tareas, instrucciones, etc. (iguales o distintas) se ejecutan en paralelo.

- Nivel de instrucción (ILP) - se ejecutan en paralelo las instrucciones de un programa. Granularidad fina.
- Nivel de bucle o hebra (Thread) - se ejecutan en paralelo distintas iteraciones de un bucle o secuencias de instrucciones de un programa. Granularidad fina/media.
- Nivel de procedimiento (Proceso) - distintos procedimientos que constituyen un programa se ejecutan simultáneamente. Grano medio.
- Nivel de programa - la plataforma ejecuta en paralelo programas diferentes que pueden corresponder, o no, a una misma aplicación. Granularidad gruesa.

7.

¿Qué diferencia existe entre el protocolo de coherencia de caché MESI y MSI? ¿Qué ventajas tiene uno de ellos sobre el otro?

MESI

Es una ampliación del protocolo de invalidación de 3 estados (MSI). Refinado para aplicaciones “secuenciales” que corren en multiprocesadores. MESI añade el estado Exclusivo (E), que indica que el bloque es la única copia (exclusiva) del sistema multiprocesador y que no está modificado, ningún otro procesador tiene el bloque en la caché y la memoria principal está actualizada.

La ventaja es que, al ser exclusivo, es posible realizar una escritura o pasar al estado modificado sin ninguna transacción en el bus, al contrario que en el caso de estar en el estado compartido; pero no implica pertenencia, así que al contrario que en el estado modificado la caché no necesita responder al observar una petición de dicho bloque. En el MSI el programa cuando lee y modifica un dato tiene que generar 2 transacciones incluso en el caso de que no exista compartición (solo presente en una caché) del dato.