

Arquitectura e Ingeniería de Computadores

Julio 2009

Nombre:

Grupo:

Normas de realización:

- Las preguntas han de ser respondidas en bloques de folios independientes, incluyendo el nombre en cada uno de los folios.
- Todas las respuestas han de ser correctamente detalladas y razonadas.
- No se corregirán las respuestas a lápiz.

Pregunta 1. (1,5 ptos)

Se tiene un cauce multifuncional con 4 etapas S1-S2-S3-S4, que puede ejecutar 2 tipos de operaciones, X e Y. Las operaciones de tipo **X** recorren el cauce según la secuencia S2-S4-S3-S1-S1-S2, mientras que las del tipo **Y** lo hacen en el orden S1-S2-S1-S2-S3-S4. El bucle interno de un programa realiza intensivamente la secuencia de operaciones **XYXXYYYY**.

Se pide calcular:

- 1) (0.75 ptos) El tiempo que tardaría en ejecutar el cuerpo del bucle si la frecuencia del reloj es de 1Ghz.
- 2) (0.75 ptos) La productividad máxima del cauce para esa secuencia.

Pregunta 2. (2 ptos)

Suponga un procesador en el que todas las instrucciones pueden predicarse. Para establecer los valores de los predicados se utilizan instrucciones de comparación con el formato

(p) p1, p2 cmp.cnd rx, ry donde **cnd** es la condición que se comprueba entre **rx** y **ry** (lt, ge, eq, ne,...). Si la condición es verdadera, **p1**=1 y **p2**=0, y si es falsa, **p1**=0 y **p2**=1. La instrucción sólo se ejecuta si el valor del predicado **p** es 1 (habrá sido establecido por otra instrucción de comparación).

(a) En estas condiciones, escriba la secuencia de instrucciones máquina que implementarían el siguiente código sin utilizar ninguna instrucción de salto (excepto para el bucle) y planificar su ejecución en un procesador VLIW con 3 slots de emisión, en el que las instrucciones de comparación sólo pueden colocarse en el primero de ellos (el resto pueden aparecer en cualquier campo). Las latencias de las operaciones son de 1 ciclo para las operaciones aritméticas y almacenamientos (la siguiente instrucción puede ejecutarse en el ciclo siguiente), y de 2 ciclos para las cargas y saltos.

```
int x[100]; //32 bits
for i=99 to 0 do
    if (x[i]/4==2) then
        x[i]=8*x[i];
    else
        x[i]=0;
endfor
```

Utilizar como modo de direccionamiento Registro base + desplazamiento. Ejemplo: lw r1, 16(reg). Suponer que inicialmente: r10= dirección final de x,

$r_2=2$ y $r_0=0$. Inicializar todos los predicados. Predicar sólo las instrucciones necesarias.

(b) Desenrollar el bucle 2 veces y planificar su ejecución de forma óptima. ¿Cuál es la ganancia que se obtiene respecto del apartado a)?

Pregunta 3 (1,5 pts)

Se dispone de una arquitectura vectorial que incorpora una memoria con entrelazado superior distribuida en 6 bloques con 6 palabras cada bloque. Se almacena un vector de 12 componentes a partir de la dirección 00h con un stride de 2. A la hora de acceder a los elementos del vector, el diseñador de una aplicación duda entre utilizar un tipo de acceso S o un tipo C. Calcula los tiempos que se tardaría en acceder a los 12 elementos del vector según cada tipo de acceso.

Pregunta 4 (1,5 pts)

Supongamos que se va a ejecutar en paralelo la suma de n números en una máquina con p procesadores (p y n potencias de 2) utilizando un grafo de dependencias en forma de árbol (divide y vencerás) para las tareas. Supongamos que se tarda una unidad de tiempo en realizar una suma.

Se pide:

- 1) (0.25 pts) Dibujar el grafo de dependencias entre tareas para $n = 16$ y $p = 8$. Realizar una asignación de tareas a procesos.
- 2) (0.25 pts) Obtener el tiempo de cálculo paralelo para cualquier n y p con $n > p$.
- 3) (0.5 pts) Obtener la **ganancia**, teniendo en cuenta que al tiempo de cálculo paralelo debemos añadir el tiempo de comunicación (**sobrecarga**) del algoritmo, suponiendo que las comunicaciones en un nivel del árbol se pueden realizar en paralelo en la red en un número de unidades de tiempo igual al número de datos que recibe o envía un proceso en cada nivel del grafo de tareas (tener en cuenta la asignación de tareas a procesos que se ha considerado en el apartado 1).
- 4) (0.5 pts) Obtener la complejidad de la función de **isoeficiencia**.

Pregunta 5 (1,5 pts)

- a) Un sistema multicomputador de 16 nodos utiliza una red tipo barril para conectar los nodos entre sí. Explica con qué elementos estaría conectado el nodo 1.
- b) ¿Qué algoritmo de los estudiados en la asignatura utilizarías para conseguir que una red mariposa fuera totalmente adaptativa y libre de ciclos? **Razona** tu respuesta.
- c) Dibuja una red mariposa que interconecte 8 entradas con 8 salidas utilizando conmutadores 2×2 .

Pregunta 6 (2 ptos)

a)(0,5 ptos) Explique la diferencia básica entre los modelos de consistencia SC y TSO ¿Cuál ahorra más tiempo? El que ahorra más tiempo, ¿lo hace en cualquier caso, o solo en determinadas condiciones (indique si es el caso, las condiciones).

b)(0,75 ptos) Calcule el tamaño que ocupará el directorio de coherencia en un multiprocesador con memoria compartida de 64 nodos, con un protocolo de directorio centralizado de tipo mapeado completo, bloques de 32 bytes, memoria principal de 512 MiB/nodo. Suponga que cada bloque guarda adicionalmente la codificación de 4 posibles estados del mismo.

c) (0,75 ptos) Suponga que la *cache* del nodo 21 envía una petición de fallo de lectura a la memoria. El bloque en memoria tiene activado el bit de inconsistencia única. Explique **detalladamente** toda la secuencia de eventos hasta que la cache recibe el bloque.