1 Un ordenador con 2 núcleos y 6 BG de RAM y una Workstation con 4 núcleos y 32 GB de RAM ejecutan 5 veces el mismo benchmark monohilo. La tabla muestra sus tiempos de respuesta en segundos:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | T1 | T2 | T3 | T4 | T5 |
| Laptop | 20.87 | 21.15 | 20.98 | 21.45 | 22.06 |
| Workstation | 4.15 | 4.19 | 4.65 | 4.37 | 3.99 |

1. ¿Rendimiento de cada computador usado?
2. ¿Aceleración de la Workstation con respecto al laptop?
3. Una versión multihilo del benchmark se ejecuta. Usa 8 hilos. El tiempo de respuesta del benchmark en cada computador es el mostrado en la tabla anterior dividido por el número de hilos que pueden correr de forma concurrente. ¿Cuál es la aceleración del Workstation en comparación al ordenador?

2 Dada una CPU segmentada de 4 etapas con un período de 0.5 ns:

1. ¿Cuál es el máximo rendimiento expresado en MIPS?
2. ¿Si la CPU fuese superescalar con una anchura de 3, cuál sería el máximo rendimiento expresado en MIPS?

3 El siguiente código corre en una microarquitectura MIP64 segmentada con las siguientes mejoras: unidad mul/div de 3 ciclos y evaluación agresiva de saltos en ID. No soporta excepciones precisas.

Ori r5, r0, 5

Xor r1, r4, r5

Bnez r5, label

Slt r4, r8, r7

Label:

Dmul r3, r8, r6

Andi r3, r0, 7

1. Escribe las dependencias de datos

RAW

WAW

WAR

1. Escribe las detenciones

RAW

WAW

Estructurales

Control

1. ¿Cuántos ciclos de reloj requiere para su ejecución?¿Cuál es el CPI ignorando el transitorio inicial?
2. ¿Si todos los reenvíos se habilitan cuáles se activarán?
3. Si se implementa el renombrado de registros, y se dispone de una cola FIFO desde rr32 hasta rr63, escribe qué registros se pueden renombrar.

4 Una arquitectura MIP64 con evaluación clásica de saltos(MEM), predicción de saltos siempre no tomado y no se permiten ejecuciones fuera de orden. Se ejecuta el siguiente código:

1. Rellena las filas

Ori r1, r0, 30

Start:

Beqz r1, end

Dsub r2, r3, r4

Daddi r1, r1, -3

J start

End:

Sd r2, 100(r0)

Xor r2, r2, r2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Ori r1, r0, 30 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Beqz r1, end |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. ¿Cuántas veces acierta el programa? ¿Cuántas falla?

1 Un ordenador portátil con 4 núcleos y 8 GB de RAM y una estación de trabajo de 10 núcleos y 64 GB ejecutan 5 veces dos benchmarks, B1 y B2. La siguiente tabla muestra los tiempos de ejecución en segundos.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B1 | | | | | B2 | | | | |
| T1 | T2 | T3 | T4 | T5 | T1 | T2 | T3 | T4 | T5 |
| Laptop | 21.37 | 22.04 | 23.05 | 21.98 | 22.08 | 2.31 | 2.27 | 2.14 | 2.16 | 2.34 |
| Wrkst. | 15.64 | 14.87 | 14.25 | 14.65 | 15.01 | 1.11 | 1.12 | 1.14 | 1.17 | 1.51 |

1. ¿Cuál es el rendimiento de cada computador utilizando la productividad?
2. ¿Cuál es la aceleración agregada de la estación de trabajo con respecto al portátil?
3. ¿Cuál es la aceleración agregada de la estación de trabajo respecto al ordenador portátil?

2 Sam y Sally refactorizan cada 1 una copia del mismo programa. Sam consigue acelerar 75 veces el 25% del programa, mientras que sale y consigue acelerar 25 veces el 75% del programa.

1. ¿Cuál es la aceleración global que logra Sam en el programa? Escribe la expresión matemática utilizada.
2. ¿Cuál es la aceleración que logra Sally en el programa?. Escribe la expresión matemática que utilices para calcularla.

3 Se ejecuta el siguiente programa MIPS64 sobre una microarquitectura segmentada básica cuya única mejora es una unidad no segmentada de multiplicación/división de enteros de 5 ciclos. Las excepciones precisas no están soportadas.

J label

Dadd r4, r1, r5

Label:

Xor r5, r5, r5

Bne r5, r0, label

Dmul r3, r8, r7

And r3, r3, r9

1. Enumera las dependencias de datos que existen en el programa especificando tipo junto con las instrucciones y registros involucrados.

RAW

WAW

WAR

1. Indica las detenciones expresadas en ciclos de reloj.

RAW

WAW

Estructural

Control

1. ¿Cuántos ciclos de reloj son necesarios para ejecutar el código anterior?. ¿Cuál es el CPI ignorando el transitorio inicial?
2. Si todas las rutas de reenvío estuviesen implementadas en esta micro arquitectura ¿qué rutas se activarían al ejecutar el programa anterior?
3. Si se implementa el renombrado de registros siguiendo la cola fifo rr32 rr63 ¿qué registros se cambiarían?

4 Una microarquitectura MIPS64 implementa evaluación clásica de saltos (en la etapa MEM), predicción de saltos siempre no tomado y sin rutas de reenvío. Se ejecuta el siguiente código:

Daddi r1, r0, 27

Startloop:

Beqz r1, exitloop

Daddi r1, r1, -3

Xor r3, r7, r6

J startloop

Exitloop:

Sd r3, 100(r0)

Xor r3, r3, r3

1. Rellena las filas

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Daddi r1, r0, 27 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Beqz r1, exitloop |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. Teniendo en cuenta las dos instrucciones de salto de programa ¿cuántas veces en total acierta la predicción y cuántas veces falla?