Exámenes

2022-UT2A (2.1,2.2,2.3) Prueba de seguimiento (Castellano)

Volver a la Lista de Exámenes

Parte 1 de 6 / 3.5 Puntos

Preguntas 1 de 13

0.7 Puntos

En el procesador MIPS segmentado, las instrucciones de carga insertan 1 ciclo de parada si la instrucción siguiente consume en su etapa EX el dato leído de la memoria. Si el compilador, en esos casos, coloca instrucciones NOP entre ambas instrucciones, el tiempo de ejecución del programa (aumenta/disminuye/no cambia) ×disminuye

Respuesta correcta: no cambia

Preguntas 2 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Teniendo en cuenta la ruta de datos segmentada del procesador MIPS en las etapas IF, ID, EX, M y WB, y teniendo en cuenta que el ciclo de reloj es de 10 ns, indica qué respuesta es CIERTA:

- ✓ A. La aceleración que se obtendría, en comparación a la ruta de datos sin segmentar, es de 5, independientemente de la duración de cada etapa
- ✓ B. El tiempo de lectura o escritura en el banco de registros no puede ser inferior a 10 ns
- ✓ C. Todas las etapas deben tener un retardo idéntico a 10 ns
- ✓ D. La suma de retardos del registro de segmentación y la etapa más lenta no puede superar los 10 ns

Respuesta correcta: D

Preguntas 3 de 13

0.7 Puntos

Indica en cuáles de los siguientes fragmentos de código se aplicaría un cortocircuito WBaEX:

A. ----------

```
dadd r1, r2, r3
dadd r4, r1, r5
dsub r6, r4, r7
```

B. ----------

```
dadd r1,r2,r3
and r20,r2,r3
ld r3,100(r1)
```

C. ----------

```
l.d f0, 0(r0)
add r1, r1, r2
s.d f0, 20(r0)
```

D. ----------

```
l.d f0, 0(r0)
s.d f0, 20(r0)
add r1, r1, r2
```

Respuesta correcta: B, C

Preguntas 4 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Indica el número de ciclos de parada y el cortocircuito que aplicaría el procesador MIPS segmentado para resolver los riesgos de datos generados por la secuencia de instrucciones mostrada:

```
ld r1,100(r10)
sd r1,200(r11)
```

- ✓ A. 0 stalls, WBaEX
- ✓ B. 0 stalls, MEMaMEM
- ✓ C. 1 stalls, WBaMEM
- ✓ D. 0 stalls, WBaMEM

Respuesta correcta: D

Preguntas 5 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Si un procesador soporta un comportamiento preciso frente a las excepciones:

- ✓ A. Se puede identificar la instrucción causante de la excepción.
- ✓ B. Las instrucciones posteriores a la excepción terminan correctamente.
- ✓ C. Todas las demás respuestas son correctas.
- ✓ D. Las instrucciones anteriores a la excepción se cancelan.

Respuesta correcta: A

Parte 2 de 6 / 2.1 Puntos

Preguntas 6 de 13

0.7 Puntos

Dado el fragmento de código MIPS que se muestra a continuación:

```
1 bnez r1,loop
2 l.d f0,100(r10)
3 add.d f4,f0,f2
4 s.d f4,100(r10)
5 l.d f0,200(r10)
6 sub.d f4,f0,f3
7 s.d f4,200(r10)
```

relaciona cada par de instrucciones con un tipo de dependencia:

A. i3 y i6
B. i2 y i3
C. i4 y i6
D. i1 y i6

✓	B	1. Dependencia de datos
✓	C	2. Antidependencia
✓	A	3. Dependencia de salida
✓	D	4. Dependencia de control

Respuesta correcta: 1:B, 2:C, 3:A, 4:D

Preguntas 7 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Asumiendo una ruta de datos con un operador multiciclo de multiplicación en coma flotante con tiempo de evaluación (o latencia) de 4 ciclos y tasa de iniciación (initiation rate) de ¼ de ciclo, podemos afirmar que:

- ✓ A. El operador no está segmentado.
- ✓ B. El operador podrá ejecutar al mismo tiempo cuatro instrucciones, pero cada una de ellas en una etapa distinta.
- ✓ C. El operador está segmentado y permite introducir a ejecución una operación cada cuatro ciclos.
- ✓ D. El operador no está segmentado y tiene un tiempo de ejecución de un cuarto de ciclo.

Respuesta correcta: A

Preguntas 8 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Indica el número de ciclos de parada que aplicaría el procesador MIPS segmentado para resolver el riesgo generado por la secuencia de instrucciones mostrada. Considera que las latencias del multiplicador y del sumador son 4 y 3, respectivamente:

```
mul.d f0,f1,f2

add.d f0,f3,f4
```

- ✓ A. 4 stalls
- ✓ B. 1 stalls
- ✓ C. 0 stalls
- ✓ D. 3 stalls

Respuesta correcta: B

Parte 3 de 6 / 1.4 Puntos

Preguntas 9 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Un predictor de dos niveles de tipo gselect con dos bits de historia global, obtiene la predicción de una instrucción de salto:

- ✓ A. teniendo en cuenta el comportamiento del salto en cuestión y de la última instrucción de salto ejecutada
- ✓ B. necesita dos tablas que almacenan predicciones de 2 bits.
- ✓ C. teniendo en cuenta el comportamiento del salto en cuestión y de las dos últimas instrucciones de salto ejecutadas
- ✓ D. teniendo en cuenta únicamente el comportamiento del salto en cuestión

Respuesta correcta: C

Preguntas 10 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Los predictores P1 y P2 de un predictor híbrido se conectan, correspondientemente, a las entradas 0 y 1 del multiplexor controlado por el predictor de selección (*tournament predictor*) que se implementa como un contador de saturación de dos bits. El predictor se selecciona con el bit de mayor peso del contador. El funcionamiento del predictor híbrido es el siguiente:

- ✓ A. Se incrementa el contador cada vez que la predicción es saltar en cualquiera de los dos predictores (P1 y P2).

- ✓ B. Se incrementa el contador siempre que acierta el predictor P2 y se decrementa siempre que acierta el predictor P1.
- ✓ C. Se decrementa el contador cuando acierta el predictor P1 y falla el predictor P2.
- ✓ D. Se incrementa el contador cada vez que la predicción es saltar en el predictor P1.

Respuesta correcta: C

Parte 4 de 6 / 1.0 Puntos

Preguntas 11 de 13

1.0 Puntos

Teniendo en cuenta la ruta de datos del procesador MIPS segmentada en cinco etapas (IF: etapa 1 del ciclo de instrucción, ID: etapa 2, EX: etapa 3, M: etapa 4, WB: etapa 5), que aplica todos los cortocircuitos posibles para resolver conflictos de datos, que resuelve los conflictos de control mediante ciclos de parada, que calcula la condición de salto en la etapa 2 del ciclo de instrucción y que modifica el PC en la etapa 2, y que no tiene ningún conflicto estructural, calcula el CPI para un alto número de iteraciones del bucle en el siguiente código:

```
loop: ld r3, 0(r2)
      ld r4, 0(r3)
      daddi r2, r2, 8
      dadd r1, r1, r4
      sd r4, 1024(r3)
      daddi r10, r10, -1
      bnez r10, loop
      sd r1, 0(r11)
      <sgte+1>
      <sgte+2>
      <sgte+3>
```

CPI = ✓1.29

Respuesta correcta: 1.29

Parte 5 de 6 / 1.0 Puntos

Preguntas 12 de 13

1.0 Puntos

Sea el siguiente código que se ejecuta en un procesador MIPS:

```
loop:       l.d f0, 0(r1)
            mul.d f1, f0, f0
            s.d f1, 0(r2)
            dadd r1, r1, 8
            dadd r3, r3, -8
            dadd r2, r2, 8
            bnez r3, loop
```

El procesador resuelve los riesgos de datos mediante ciclos de parada y cortocircuitos, mientras que los riesgos de control los resuelve con la técnica *predict-not-taken*, actualizando el PC en la etapa ID. El procesador dispone de un multiplicador segmentado multiciclo con Tev=5.

Aplica la técnica de loop unrolling con el fin de reducir al máximo los ciclos de parada por riesgos de datos, así como incrementar al mínimo el número de instrucciones. ¿Cuántas iteraciones del bucle original deberían incluirse en el nuevo cuerpo del bucle? ✓4

Respuesta correcta: 4

Parte 6 de 6 / 1.0 Puntos

Preguntas 13 de 13

1.0 Puntos

Sea el siguiente código en ensamblador:

```
nozero:     li t0, 13         # Número de elementos del vector
            li v0, 0          # contador inicial = 0
            li t1, V          # dirección vector V
loop:       lw t2, 0(t1)      # lectura V[i]
            addi t0, t0, -1   # Decrementa elementos vector
            bnez t2, sigue    # Si V[i] es distinto de cero salta
            addi v0, v0, 1    # Incrementa contador
sigue:      addi t1, t1, 4    # Incrementa dirección vector V
            bnez t0, loop     # Siguiente iteración
```

Dicho código implementa la función *nozero* que calcula el número de elementos de un vector de 13 elementos con valor igual a cero.

El código se ejecuta en un procesador segmentado de 5 etapas el cual resuelve todos los riesgos de datos con cortocircuitos. El procesador implementa un BTB con un predictor de dos bits con saturación. La tabla tiene 16 entradas e inicialmente está vacía. En ausencia de historia del salto se utiliza *predict-not-taken*.

Cuando la información del salto se almacena por primera vez en la BTB, el estado del predictor se pone a "00" (*Strongly Not Taken*) si el salto no es efectivo y a "11" (*Strongly Taken*) en caso contrario. Un fallo de predicción ocasiona la inserción de 2 ciclos de parada.

Indica cuantos ciclos de penalización en total introducirá cada instrucción de salto en la ejecución del código anterior para el caso de un vector que contenga elementos con los valores “0 0 0 0 0 0 0 0 0 0 1 0 0”

a) Ciclos de penalización `bnez t2, sigue`: ✓2 ciclos

b) Ciclos de penalización `bnez t0, loop`: ✓4 ciclos

Respuesta correcta: 2, 4