

## 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) ✓ no cambia

**Respuesta correcta:** no cambia

Preguntas 2 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. -----

```
l.d f0, 0(r0)
```

```
add r1, r1, r2
```

```
s.d f0, 20(r0)
```

C. -----

```
l.d f0, 0(r0)
```

```
s.d f0, 20(r0)
```

```
add r1, r1, r2
```

D. -----

```
dadd r1,r2,r3
```

✓

```
and r20,r2,r3
```

```
ld r3,100(r1)
```

**Respuesta correcta:** B, D

Preguntas 3 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. El tiempo de lectura o escritura en el banco de registros no puede ser inferior a 10 ns
- ✓ B. La suma de retardos del registro de segmentación y la etapa más lenta no puede superar los 10 ns
- ✓ C. La aceleración que se obtendría, en comparación a la ruta de datos sin segmentar, es de 5, independientemente de la duración de cada etapa
- ✓ D. Todas las etapas deben tener un retardo idéntico a 10 ns

**Respuesta correcta:** B

Preguntas 4 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Indica cuál de los siguientes casos se ocasiona un riesgo estructural:

- ✓ A. Una instrucción accede a memoria para leer o escribir un dato y el procesador tiene una memoria cache unificada de datos e instrucciones.
- ✓ B. Dos instrucciones acceden en el mismo ciclo de reloj al banco de registros, una para realizar una escritura de un resultado y la otra para leer operandos. El banco de registros tiene un puerto de lectura por operando y un puerto de escritura.
- ✓ C. Dos instrucciones aritméticas calculan su resultado en la misma etapa en ciclos consecutivos.
- ✓ D. Una instrucción accede a memoria para leer o escribir un dato y el procesador tiene memorias cache separadas de datos e instrucciones.

**Respuesta correcta:** A

Preguntas 5 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, WBaMEM
- ✓ C. 0 stalls, MEMaMEM
- ✓ D. 1 stalls, WBaMEM

**Respuesta correcta:** B

## Parte 2 de 6 / 2.1 Puntos

Preguntas 6 de 13

0.7 Puntos

Indica qué tipo de dependencia existe (**datos, antidependencia, salida**) en los siguientes supuestos:

l.d f0, 0(r1)

...

l.d f0, 0(r2)

dependencia: ☒ salida

dadd r10, r1, r2

dadd r1, r2, r3

dependencia: ☒ antidependencia

dadd r10, r1, r2

dadd r3, r10, r2

dependencia: ☒ datos

s.d f0, X(r1)

l.d f1, X(r1)

dependencia: ☒ datos

**Respuesta correcta:** salida, antidependencia, datos, datos

Preguntas 7 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Asumiendo una ruta de datos segmentada con operadores multiciclo de multiplicación (Latencia=4, IR=1) y de suma (Latencia=2, IR=1) y con banco de registros de enteros y coma flotante separados, indica qué afirmación es CIERTA:

- ☒ A. La ruta de datos no tiene necesidad de insertar ciclos de parada ya que todos los riesgos se pueden resolver por cortocircuitos.
- ☒ B. Para evitar riesgos WAW, deben insertarse ciclos de parada.

- ✓ C. Todos los riesgos estructurales se pueden resolver al utilizar dos bancos de registros diferenciados de coma flotante y de enteros.
- ✓ D. La ruta de datos podría introducir riesgos estructurales en la etapa WB entre una instrucción que escriba en los registros enteros y otra que escriba en los de coma flotante.

**Respuesta correcta:** B

Preguntas 8 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. Considera que las latencias del multiplicador y del sumador son 7 y 4, respectivamente:

```
mul.d f0,f1,f2
```

```
add.d f0,f0,f3
```

- ✓ A. 0 stalls, WBaA1
- ✓ B. 7 stalls, WBaA1
- ✓ C. 6 stalls, WBaA1
- ✓ D. 6 stalls, M7aA1

**Respuesta correcta:** C

### Parte 3 de 6 / 1.4 Puntos

Preguntas 9 de 13

0.7 Puntos. Puntos descontados por fallo: 0.21

Indica cual de las siguientes afirmaciones es CIERTA:

- ✓ A. Un predictor de saltos de dos bits acierta el doble de veces que un predictor de un bit.

- ✓ B. Un predictor de saltos de dos bits con histéresis, después de dos saltos consecutivos no efectivos de la misma instrucción predecirá salto no efectivo para esa instrucción.
- ✓ C. Un predictor de saltos de dos bits con saturación predice salto efectivo solamente cuando el contador satura.
- ✓ D. Un predictor de saltos actualiza el contador (estado) en el momento de hacer la predicción.

**Respuesta correcta:** B

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

**Respuesta correcta:** A

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

CPI = ✓ 1.43

**Respuesta correcta:** 1.43

## 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  $T_{ev}=3$ .

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? ☒ 2

**Respuesta correcta:** 2

## 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 0 0"

a) Ciclos de penalización `bnez t2, sigue`: ☒ 0 ciclos

b) Ciclos de penalización `bnez t0, loop`: ☐ 24 ciclos



**Respuesta correcta:** 0, 4