

## Exámenes

### UT2 (2.4,2.5) - Prueba de seguimiento

[Tabla de Contenidos](#)

Tiempo restante: 0:11:37

[Ocultar/Mostrar el tiempo restante](#)

Parte 1 de 4 -

Preguntas 1 de 8

1 Puntos

Indica cuál de las siguientes afirmaciones es **falsa**:☐

A.

Cuando la instrucción llega a la etapa WB, difunde su resultado a través del bus de datos común para que todas las estaciones de reserva, buffers y registros que requieran de dicho valor puedan recuperarlo.

☐

B.

La planificación dinámica de instrucciones basada en el algoritmo de Tomasulo con especulación hardware sólo afecta a la ejecución de instrucciones multiciclo, como las de aritmética en coma flotante.

☐

C.

Las estaciones de reserva almacenan las instrucciones desde que se lanzan a ejecución (I) hasta que se confirman (C).

☐

D.

Una instrucción dependiente solo podrá obtener sus operandos de los registros fuente que utilice, y si éstos estuvieran reservados por otra instrucción, anotarse la marca que éstos contengan.

Preguntas 2 de 8

1 Puntos

Gestionar dinámicamente las instrucciones a ejecutar conlleva el mantenimiento de un grafo de dependencias. Se añaden nuevas entradas al grafo cuando las instrucciones pasan por la etapa  , mientras que se eliminan cuando pasan por la etapa .

## Preguntas 3 de 8

1 Puntos

Indica cuales de las siguientes afirmaciones son correctas:

☐

A. Las instrucciones especulativas pueden generar excepciones durante su ejecución, aunque éstas se tratarán cuando las instrucciones sean confirmadas.

☐

B. Las instrucciones especulativas pueden ejecutarse fuera de orden.

☐

C. A la fase Commit pueden llegar instrucciones especuladas incorrectamente, pero éstas se cancelan.

☐

D. Una de las ventajas de la especulación hardware es que permite la confirmación de instrucciones fuera de orden.

Parte 2 de 4 -

## Preguntas 4 de 8

1 Puntos

Indica cuales de las siguientes afirmaciones son correctas:

☐

A. Los procesadores multihilo con multithreading de grano grueso no retardan la ejecución de cada hilo individual y permiten ocular stalls tanto de alta como de baja latencia.

☐

B. Ninguna de las anteriores

☐

C. Los procesadores multihilo mejoran el ILP de los procesadores superescalares ejecutando en paralelo múltiples flujos de instrucciones.

☐

D. Los procesadores multihilo conmutan de hilo en caso de fallo de cache.

☐

E. Los procesadores multihilo con multithreading simultáneo tratan de utilizar todos los recursos disponibles, aunque la ejecución simultánea de instrucciones de distintos hilos exige de algunos recursos adicionales para distinguir el estado de los hilos.

☐

F. Los procesadores multihilo con multithreading de grano fino conmutan entre hilos cada ciclo de reloj, con lo que incrementa la productividad, aunque se retarda la ejecución de cada hilo individual.

## Preguntas 5 de 8

1 Puntos

Indica cuales de las siguientes afirmaciones es correcta:

☐

A. Los procesadores supersegmentados necesitan replicar los operadores y las memorias para poderlos utilizar en las nuevas etapas que se introducen en sus pipelines.

☐

B. Los procesadores supersegmentados ofrecen una mayor frecuencia de funcionamiento, aunque ésta puede verse limitada por el desfase máximo que presente la señal de reloj.

☐

C. Los procesadores supersegmentados super-segmentan el ciclo base del procesador escalar correspondiente en un factor  $p$ , que define su grado de supersegmentación.

☐

D. Los procesadores supersegmentados mejoran las prestaciones de sus homólogos segmentados aumentando su frecuencia de funcionamiento y , por tanto, reduciendo el número de ciclos promedio requeridos para la ejecución de las instrucciones.

## Preguntas 6 de 8

1 Puntos. Puntos descontados por fallo: 0.3

Los procesadores VLIW aumentan la complejidad de las instrucciones máquina y el trabajo que éstas realizan con el objetivo de reducir el tiempo de ejecución de los programas.

☒ Verdadero☐ Falso[Borra selección](#)

Parte 3 de 4 -

## Preguntas 7 de 8

2 Puntos

La ejecución de un bucle en un procesador que aplica gestión dinámica de instrucciones con especulación hardware proporciona el siguiente diagrama instrucciones-tiempo:

PC	Instruc.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
.text	dadd r1,r0,r0	IF	I	E1	WB	C																
.text + 04	ld r2,n(r0)		IF	I	AC	L1	L2	WB	C													
.text + 08	ld f0,a(r0)			IF	I	AC	L1	L2	WB	C												
loop	ld f2,x(r1)				IF	I	AC	L1	L2	WB	C											
loop + 04	ld f4,y(r1)					IF	I	AC	L1	L2	WB	C										
loop + 08	mul.d f2,f2,f0						IF	I			M1	M2	M3	M4	WB	C						
loop + 12	add.d f4,f4,f2							IF	I							A1	A2	WB	C			
loop + 16	s.d f4,z(r1)								IF	I	AC									C	L1	L2
loop + 20	dadd r1,r1,#8									IF	I	E1	WB								C	
loop + 24	bne r1,r2,loop										IF	I		E1	-	WB						C
loop	ld f2,x(r1)											IF	I	AC	L1	L2	WB					
loop + 04	ld f4,y(r1)												IF	I	AC	L1	L2	-	WB			
loop + 08	mul.d f2,f2,f0													IF	I			M1	M2	M3	M4	WB
loop + 12	add.d f4,f4,f2														IF	I						
loop + 16	s.d f4,z(r1)															IF	I	AC				
loop + 20	dadd r1,r1,#8																IF	I	E1	WB		
loop + 24	bne r1,r2,loop																	IF	I		E1	-
loop	ld f2,x(r1)																		IF	I	AC	L1
loop + 04	ld f4,y(r1)																			IF	I	AC
loop + 08	mul.d f2,f2,f0																				IF	I
loop + 12	add.d f4,f4,f2																					IF

El código asociado a cada iteración del bucle comienza en la posición *loop* y termina en la posición *loop+24*. A la vista del diagrama instrucciones-tiempo suministrado, responde a las siguientes preguntas:

- ¿Cuántos ciclos tarda el bucle en ejecutarse si el predictor no falla?  ciclos
- ¿Cuántos ciclos de penalización en la ejecución del bucle introducirá un fallo del predictor?  ciclos
- Asumiendo que se utiliza un BTB que asocia un predictor de 2 bits con histéresis a cada salto, y que antes de ejecutar el bucle, la instrucción de salto del mismo no está en el BTB, ¿cuál será el CPI promedio (expresado con 3 decimales) resultante de procesar unos vectores X e Y de 64 elementos cada uno? El CPI promedio será de  ciclos/instrucción.

Parte 4 de 4 -

Preguntas 8 de 8

2 Puntos

Se ha ejecutado un bucle en un procesador superescalar de 2 vías que aplica planificación dinámica de instrucciones basada en el algoritmo de Tomasulo con especulación. El lanzamiento de las instrucciones se realiza en orden y alineado. El bucle ejecutado comienza en la posición de memoria *loop*. El diagrama instrucciones-tiempo resultante de la ejecución de la primera iteración del bucle es el siguiente:

PC	Instruc.	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>
.text	dsub r1,r1,#8	IF	I	E1	WB	C		
.text + 04	ld r1,n(r0)	IF	I	AC	L1	L2		
.text + 08	l.d f0,a(r0)		IF	I	AC			
loop	l.d f2,x(r1)		IF	I				
loop + 04	mul.d f2,f2,f0			IF	I			
loop + 08	add.d f2,f2,f0			IF	I			
loop + 12	s.d f2,y(r1)				IF	I		
loop + 16	dsub r1,r1,#8				IF	I		
loop + 20	bnez r1,loop					IF		
loop + 24	trap 0					IF		

Las características de las unidades funcionales de la unidad de ejecución de instrucciones son las siguientes:

Unidad	Nº Operadores	Latencia	Segmentado	Características	Etapas
Carga/Almacenamiento	1	2	No	3 TL, 3 TE	L1, L2
Sumador/Restador c.f.	1	2	No	3 ER	A1, A2
Multiplicador c.f.	1	4	Sí	2 ER	M1, M2, M3, M4
Enteros/Saltos	1	1	No	5 ER	E1

Se pide rellenar el diagrama instrucciones-tiempo que se presenta a continuación para los ciclos 6 y 7. En el diagrama aparecen las instrucciones que ya hay en ejecución, pero se deben incluir aquellas que se busquen en memoria durante los dos ciclos considerados. En caso de no tener que hacer nada en un ciclo para una instrucción **NO DEJES NINGUNA CELDA DE LA TABLA EN BLANCO utiliza un guión (-) para señalar dicha situación.**

PC	Instrucciones	6	7
.text	dsub r1, r1, #8	-	-
.text+04	ld r1, n(r0)	WB	C
.text+08	l.d f0, a(r0)	L1	L2
loop	l.d f2, x(r1)	-	-
loop+04	mul.d f2, f2, f0	-	-

loop+08	add.d f2, f2, f0	-	-
loop+12	s.d f2, y(r1)	AC	-
loop+16	dsub r1, r1, #8	E1	WB
loop+20	bnez r1, loop	I	-
loop+24	trap 0	I	-
loop	l.d f2, x(r1)	IF	I
loop+04	mul.d f2, f2, f0	IF	I
loop+08	add.d f2, f2, f0	-	IF
loop+12	s.d f2, y(r1)	-	IF



- [PoliformaT](#)
- [UPV](#)
- [Powered by Sakai](#)
- Copyright 2003-2018 The Sakai Foundation. All rights reserved. Portions of Sakai are copyrighted by other parties as described in the Acknowledgments screen.