

Heroe-2-Acremoved.pdf



user_2749634



Arquitectura de Computadores



2º Grado en Ingeniería Informática del Software



**Escuela de Ingeniería Informática
Universidad de Oviedo**



Escuela de
organización
Industrial

La mejor escuela de negocios en
energía, sostenibilidad y medio
ambiente de España.

Más información
www.eoi.es

Formamos
talento para un futuro
Sostenible



100% Empleabilidad



Modalidad: Presencial u online



**Programa de Becas,
Bonificaciones y Descuentos**

CORRIGE

2

3
PROTEGE

Tu ex no te dejó las cosas claras, pero nosotros la rutina sí.



Examen 2

jueves, 27 de octubre de 2022 11:35

A

Arquitectura de Computadores

2020

Instrucciones generales para la realización de este examen

La respuesta debe escribirse en el hueco existente a continuación de cada pregunta con letra clara. Cada respuesta incorrecta, ilegible o vacía no suma ni resta. En el caso de preguntas teóricas se valorará la capacidad de síntesis.

- 1 Un ordenador portátil con 2 núcleos y 4 GB RAM y una estación de trabajo con 4 núcleos y 16 GB RAM ejecutan cinco veces el mismo benchmark monohilo. La tabla siguiente muestra el tiempo de respuesta de cada ejecución, en segundos.

	t_1	t_2	t_3	t_4	t_5	Media
Laptop	17.45	18.23	18.11	17.37	17.05	17,64
Workstation	11.23	11.18	10.98	11.35	11.07	11,16

a — (0.25 puntos) ¿Cuál es el rendimiento de cada computador utilizando la productividad?

Laptop: $P = \frac{1}{17,64} = 0,057$ Workstation: $P = \frac{1}{11,16} = 0,090$

b — (0.5 puntos) ¿Cuál es la aceleración de la estación de trabajo respecto al ordenador portátil?

$A = \frac{0,090}{0,057} = 1,58$

c — (0.75 puntos) Se ejecuta ahora una versión multihilo del benchmark anterior, que utiliza 4 hilos. El tiempo que tarda este benchmark en ejecutarse en cada sistema es el de la tabla anterior dividido por el número de hilos que se pueden ejecutar concurrentemente en el sistema. En este nuevo escenario, ¿cuál es la aceleración de la estación de trabajo respecto al ordenador portátil?

$A = \frac{0,090 \cdot 4}{0,057 \cdot 2} = 3,16$

- 2 La compresión de una película de vídeo en formato 4K con el codec H.265 lleva 8 horas en un PC con un procesador Intel i5-8400 y una unidad de disco SSD Samsung Evo 860. Se sabe que el proceso de codificación usa la CPU el 65% del tiempo y el acceso al disco el 35% restante.

a — (1 punto) ¿En cuánto tiempo se podría codificar el mismo vídeo si el procesador se sustituye por un Intel i9-9900K cuyo rendimiento es 2.11 veces mayor? ¿Y si el disco se sustituyera por un Samsung EVO 960 que es 7.11 veces más rápido? Responder en minutos.

Con nueva CPU: $8 \cdot \left((1 - 0,65) + \frac{0,65}{2,11} \right)$ Con nuevo disco: $8 \cdot \left((1 - 0,35) + \frac{0,35}{7,11} \right) = 5,59 \text{ horas} \cdot 60 = 335,4 \text{ minutos}$
 $= 5,26 \text{ horas} \cdot 60 = 315,6 \text{ minutos}$

- 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 2 ciclos. Las excepciones precisas no están soportadas.

```

1 bne r3, r3, dest
2 xori r1, r3, 2
3 dest:
4 dmul r5, r8, r1
5 ld r8, 80(r6)
6 daddi r5, r8, -1

```

1 r3 r32 r5 r33 r8 r34 r5 r35

a — (1 punto) Enumera las dependencias de datos que existen en el programa, especificando tipo junto con las instrucciones y registros involucrados. Ejemplo de respuesta: RAW => daddi, ori : r7 // dsub, xori : r3

RAW => xori, dmul : r1 ld, daddi : r8

WAW => dmul, daddi : r5

WAR => dmul, ld : r8

b — (1 punto) Indica las duraciones de las detenciones expresadas en ciclos de reloj. Para los tipos RAW, WAW y estructural debe indicarse además la instrucción que se detiene, mientras que para el tipo control la instrucción que la provoca. Si un tipo de detención no aparece, indicarlo con N/A. Si hay varias detenciones del mismo tipo deben indicarse todas.

Ejemplo de respuesta, RAW: xor 1 ciclo // andi 2 ciclos

RAW: dmul : 2 ciclos daddi : 2 ciclos

WAW: N/A

Estructural: ld : 1 ciclo

Control: xor : 3 ciclos

LIMPIA

TU RUTINA ANTI-IMPERFECCIONES

GARNIER
PureActive

A

A

c — (1 punto) ¿Cuántos ciclos de reloj son necesarios para ejecutar el código anterior? ¿Cuál es el CPI ignorando el transitorio inicial? Indica la expresión matemática utilizada para obtener el CPI.

Ciclos: 17 CPI: $\frac{17-4}{5} = 2.6$

d — (0.5 puntos) Si todas las rutas de reenvío estuviesen implementadas en esta microarquitectura, ¿qué rutas se activarían al ejecutar el programa anterior? Ejemplo de respuesta: Salida EX daddi → Entrada EX dmul.

Salida EX ori → Entrada EX dmul
Salida MEM ld → Entrada EX daddi

e — (1 punto) Si se implementase renombrado de registros en esta microarquitectura, ¿qué registros arquitectónicos cambiarían de registro físico al final de la ejecución del programa y a qué registro físico estarían asignados? Para el renombrado se dispone de una cola FIFO que originalmente contiene los registros rr32 a rr63 y a los que se van añadiendo los registros disponibles
Ejemplo de respuesta para esta pregunta: Registro r7 asignado a rr43.

r1 → rr32
r3 → rr33
r8 → rr34

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

```
1 A80C ori r13, r0, 40
2 loop:
3 A810 beqz r13, endloop ; Branch on Equal Zero
4 A814 xor r14, r15, r16
5 A818 daddi r13, r13, -4 ; r13 decrement by -4
6 A81C j loop ; Jump to loop
7 endloop:
8 A820 sd r14, 100(r0)
9 A824 xor r14, r14, r14
```

a — (1.5 puntos) Rellena todas las filas del siguiente cronograma con la evolución de las primeras instrucciones del programa sobre el pipeline.

Instrucción \ Ciclo	1	2	3	4	5	6	7	8	9	10	11	12	13	14
ori r13, r0, 40	IF	ID	EX	MEM	WB									
beqz r13, endloop		IF	ID	ID	ID	EX	MEM	WB						
xor			IF	ID	EX	MEM	WB							
daddi					IF	ID	EX	MEM	WB					
j						IF	ID	EX	MEM	WB				
sd							IF	ID	EX	MEM	WB			
beqz								IF	ID	EX	MEM	WB		
xor									IF	ID	EX	MEM	WB	

b — (0.5 puntos) Teniendo en cuenta las dos instrucciones de salto del programa. Para el programa completo, ¿Cuántas veces en total acierta la predicción el predictor siempre no tomado? ¿Y cuántas veces en total falla la predicción? Nota: el predictor solo se aplica a las instrucciones de salto condicional.

Núm. aciertos: 10 Núm. fallos: 1

c — (0.5 puntos) Si el predictor de saltos siempre no tomado se sustituye por un predictor dinámico de 2 bits con el valor 01 por defecto del historial (weak not taken), ¿cuántos ciclos de reloj se detendría el pipeline en la ejecución del programa completo debido a la instrucción beqz r13, endloop y a la instrucción j loop?

beqz r13, endloop: 3 j loop: 1

d — (0.5 puntos) Si el programa se ha cargado a partir de la dirección de memoria A80Ch ¿Cuáles serán los valores de las entradas en la tabla BHT+BTB correspondientes a las instrucciones de salto al finalizar la ejecución del programa?

Dirección	Destino	Historial
A810	A820	01 (Falló 1 vez)
A81C	A820	11 (en un j)