

Responde cada pregunta en una hoja distinta. Tiempo disponible: 1h45m

1. (2,5 puntos) Una empresa utiliza un computador valorado en 4000€ para la ejecución de dos programas A y B (que no se ejecutan concurrentemente).

Se monitoriza el sistema y se constata que:

- El programa A se ejecuta el 25 % del tiempo y el resto lo hace el programa B.
- El programa A dedica el 80 % de su tiempo a realizar cálculos en coma flotante, mientras que el programa B no realiza ningún cálculo de este tipo y sólo ejecuta instrucciones de enteros dedicadas a otros menesteres.
- El programa B emplea el 40 % de su tiempo de ejecución accediendo a disco, mientras que el programa A no realiza ningún acceso.

Considerando el sistema completo, se pide:

- a) Obtener la fracción de uso del disco.
 - b) Calcular la máxima aceleración que se obtendría cambiando sólo el disco del computador.
 - c) Determinar si el cambio del disco existente por otro 3 veces más rápido, y que cuesta 800 €, es interesante desde el punto de vista de la relación prestaciones–coste.
 - d) Finalmente, se añade una GPU para mejorar el rendimiento en coma flotante en un factor de 10 ¿en qué porcentaje se mejorarían las prestaciones del computador respecto a su configuración original si integráramos a la vez en el mismo un disco 3 veces más rápido y la GPU?
2. (3 puntos) Se tiene un MIPS64 segmentado funcionando a 3 GHz que ejecuta un programa P con la siguiente distribución de instrucciones:

Tipo	%	CPI
Cargas	24 %	1.2
UAL	50 %	1.5
Almacenamientos	14 %	1
Salto	12 %	1.1

Se plantea añadir nuevas instrucciones SIMD que operen simultáneamente sobre las dos porciones de 32 bits de cada registro de coma flotante de 64 bits. De esta manera, en los programas, se pueden sustituir secuencias de dos instrucciones que trabajan sobre números en coma flotante de simple precisión por una instrucción SIMD, como en los siguientes ejemplos:

Las secuencias:	son sustituidas por:
l.s f0,d(rb)	l.ps f0,d(rb)
l.s f1,d+4(rb)	
...	
add.s f2,f0,f10	add.ps f2,f0,f10
add.s f3,f0,f11	
...	
s.s f0,d(rb)	s.ps f0,d(rb)
s.s f1,d+4(rb)	

Con la incorporación de las nuevas instrucciones se ha medido que se pueden reemplazar el 80 % de las instrucciones de carga, el 40 % de las UAL y el 60 % de los almacenamientos. Sin embargo, la inclusión de las nuevas instrucciones afecta al CPI de las instrucciones de carga y UAL (tanto las iniciales como las SIMD), pasando a ser de 1.4 y 1.8, respectivamente. Adicionalmente, debido a la necesidad de decodificar nuevas instrucciones, la frecuencia de reloj se reduce en un 5 %.

Se pide:

- a) CPI medio del programa P en el procesador original.
- b) Si el programa P ejecuta n instrucciones en el procesador original, calcular su tiempo de ejecución en ciclos y en segundos.
- c) Dado un programa P con $n = 100$ instrucciones en el que se realizará la sustitución de instrucciones SIMD, indicar:
 - 1) El número de instrucciones ejecutadas de carga escalar y carga SIMD (l.ps).
 - 2) El número de instrucciones ejecutadas ALU escalar y ALU SIMD (por ejemplo, add.ps).
 - 3) El número de instrucciones ejecutadas de almacenamiento escalar y almacenamiento SIMD (s.ps).
 - 4) El número total de instrucciones ejecutadas en el procesador con instrucciones SIMD.
- d) CPI medio del programa P en el procesador con instrucciones SIMD.
- e) Tiempo de ejecución del programa P en el procesador con instrucciones SIMD, en ciclos y en segundos.
- f) La aceleración alcanzada, en su caso, al utilizar el procesador con instrucciones SIMD.

3. (2,5 puntos) El siguiente diagrama instrucciones–tiempo corresponde a una iteración intermedia de un bucle que se ejecuta en un procesador MIPS segmentado:

		30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	
loop:	ld r12,0(r1)		IF	ID	EX	ME	WB												
	ld r14,0(r2)			IF	ID	EX	ME	WB											
	dadd r12,r10,r12				IF	ID	EX	ME	WB										
	dadd r14,r12,r14					IF	ID	EX	ME	WB									
	sd r14,0(r3)						IF	ID	EX	ME	WB								
	dadd r1,r1,#8							IF	ID	EX	ME	WB							
	dadd r2,r2,#8								IF	ID	EX	ME	WB						
	dadd r3,r3,#8									IF	ID	EX	ME	WB					
	seq r5,r4,r1										IF	ID	EX	ME	WB				
	beqz r5,loop											IF	ID	EX	ME	WB			
	nop												IF	ID	X				
	nop													IF	X				
loop:	ld r12,0(r1)														IF	ID	EX	ME	WB

Responde a las siguientes preguntas:

- ¿Cómo resuelve la unidad de ejecución de instrucciones los riesgos de control? Explícalo e indica el nombre de la técnica utilizada.
 - ¿En qué etapa de segmentación actualizan los saltos el valor del PC en el procesador considerado? No te limites a indicar cuál es la etapa, explica qué indicios te llevan a dicha conclusión.
 - Se añade al procesador una BTB que emplea un predictor de dos bits con saturación. Cuando se introduce un nuevo salto en la BTB, el estado del predictor depende del comportamiento del salto. Si el salto “Salta”, el estado del predictor será el estado 11, en el que predice que el salto “Salta”, mientras que si “No Salta”, el estado del predictor será el estado 00, en el que se predice que el salto “No Salta”. Si el salto no está en la tabla, se predice que “No Salta”.
 - Indica el número de ciclos consumido en una iteración cuando el predictor acierta.
 - Indica el número de ciclos consumido en una iteración cuando el predictor falla.
 - Suponiendo que la tabla BTB está inicialmente vacía ¿cuántos ciclos requerirá el bucle para procesar un vector de 100 elementos?
 - Si la BTB se organiza como una tabla completamente asociativa y contara con 1024 entradas, ¿Cuántos bits del PC deberá almacenar cada una de las entradas del BTB?
4. (2 puntos) Indica, para los diagramas instrucciones–tiempo mostrados, qué técnica se ha utilizado para resolver los riesgos de datos (“parada”, “cortocircuito”) y los riesgos de control (“parada”, “predict-not-taken”, “salto retardado”, “predictor BTB”). No te limites a indicar el nombre de la técnica. Explica qué indicios te llevan a dicha conclusión.

- | PC | Instrucción | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------|---------------|----|----|----|----|----|----|----|----|----|----|
| .text | slt r1,r3,r0 | IF | ID | EX | ME | WB | | | | | |
| 4 | dsub r2,r0,r3 | | IF | ID | EX | ME | WB | | | | |
| 8 | bnez r1,then | | | IF | ID | EX | ME | WB | | | |
| 12 | nop | | | | IF | ID | X | | | | |
| 16 | nop | | | | | IF | X | | | | |
| then | dadd r4,r2,r5 | | | | | | IF | ID | EX | ME | WB |
- | PC | Instrucción | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|-------|---------------|----|----|----|----|----|----|----|----|----|----|----|
| .text | slt r1,r3,r0 | IF | ID | EX | ME | WB | | | | | | |
| 4 | dsub r2,r0,r3 | | IF | ID | EX | ME | WB | | | | | |
| 8 | bnez r1,then | | | IF | id | ID | EX | ME | WB | | | |
| 12 | nop | | | | | IF | ID | EX | ME | WB | | |
| 16 | nop | | | | | | IF | ID | EX | ME | WB | |
| then | dadd r4,r2,r5 | | | | | | | IF | ID | EX | ME | WB |
- | PC | Instrucción | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-------|---------------|----|----|----|----|----|----|----|----|
| .text | slt r1,r3,r0 | IF | ID | EX | ME | WB | | | |
| 4 | dsub r2,r0,r3 | | IF | ID | EX | ME | WB | | |
| 8 | bnez r1,then | | | IF | ID | EX | ME | WB | |
| then | dadd r4,r2,r5 | | | | IF | ID | EX | ME | WB |
- | PC | Instrucción | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------|---------------|----|----|----|----|----|----|----|----|----|----|
| .text | slt r1,r3,r0 | IF | ID | EX | ME | WB | | | | | |
| 4 | dsub r2,r0,r3 | | IF | ID | EX | ME | WB | | | | |
| 8 | bnez r1,then | | | IF | ID | EX | ME | WB | | | |
| 12 | nop | | | | if | if | | | | | |
| then | dadd r4,r2,r5 | | | | | | IF | ID | EX | ME | WB |