

## Pregunta 1

- a. Utilizando la fórmula de Amdahl:

$$1 / (1 - f_{tm} + f_{tm}/ac.c)$$

La  $f_{tm}$  es  $4/5$  y la aceleración de la característica sería utilizar un número infinito de núcleos. Como es un número infinito entonces  $f_{tm}/ac.c$  tiende a 0 y la ecuación quedaría de la siguiente forma:

$$1 / (1 - 4/5) = 5 \quad R/ \text{ La cota superior para la aceleración es } 5.$$

- b. Como la mejora a las operaciones de memoria es que pasa el CPI de 10 a 2, entonces habrá una mejora de  $1/5$  del tiempo, lo que significa que la aceleración de la característica es 5.

El tiempo total se calcularía de la siguiente forma:

$$\begin{aligned} t. \text{ total} &= (0,2 * IC * 5 + 0,3 * IC * 10 + 0,5 * IC * 4) LCR \\ t. \text{ total} &= IC(1 + 3 + 2) LCR \\ t. \text{ total} &= IC * 6 * LCR \end{aligned}$$

Además, el tiempo en operaciones de memoria es:  $IC * 3 * LCR$ , por lo que la  $f_{tm}$  sería:

$$\begin{aligned} f_{tm} &= \text{tiempo op. mem} / \text{tiempo total} \\ f_{tm} &= IC * 3 * LCR / IC * 6 * LCR = 1/2 \end{aligned}$$

Y, utilizando la fórmula de Amdahl:

$$\begin{aligned} ac &= 1 / (1 - 1/2 + (1/2) / 5) \\ ac &= 1 / (1 - 1/2 + 1/10) \\ ac &= 1 / 3/5 \\ ac &= 5/3 \quad R/ \text{ La aceleración del programa es de } 5/3. \end{aligned}$$

c.

Característica	¿Arquitectura?	¿Implementación?
# de instrucciones que se pueden emitir por ciclo de reloj		X
Tamaño de la caché de datos L1	X	
# de registros visibles para los programadores		X
Ciclos de retraso por fallos en la caché de datos L1		X
Presencia/ausencia de una instrucción <u>left shift</u>	X	
Ciclos de retraso por el <u>Branch</u>		X

## Pregunta 2

- a. El MTTF de cada servidor es de 100 días, de manera que el FIT para cada uno es de  $1/100$ . El FIT del sistema sería  $5 * 1/100 = 1 / 20$ , por lo que el MTTF del sistema es de 20 días. Con la fórmula del coeficiente de disponibilidad se tiene que:

$$\text{coeficiente disp.} = \text{MTTF} / \text{MTTR} + \text{MTTF}$$

$$20 / \text{MTTR} + 20 \geq 0.95$$

$$(20 / 0.95) - 20 \geq \text{MTTR}$$

$$400/19 - 20 \geq \text{MTTR}$$

$$20/19 \geq \text{MTTR}$$

R/ El tiempo máximo de reparación debería ser de 20/19 días.

- b. El FIT del sistema con cuatro servidores sería de  $4 * 1/100 = 1 / 25$ , por lo que el MTTF del sistema con cuatro servidores es de 25 días. El MTTF del sistema pasaría a ser de 45 días ( $20 + 25$ ), entonces, el tiempo de reparación de un servidor debería ser menor que 25 días para que el sistema no se caiga.

R/ El tiempo de reparación de un servidor debería ser menor que 25 días.

- c. El FIT del sistema con tres servidores es de  $3 * 1/100 = 3/100$ , entonces el MTTF del sistema con tres servidores es de 33,334 días, de manera que el MTTF del sistema en general pasaría a ser de 78,334 días.

R/ El MTTF del sistema si falla cuando se caen tres servidores es de 78,334 días.

## Pregunta 3

- a. El CPI de una instrucción vendría dado por:

$$6 + \text{retraso prom. fetch instrucc.} + \text{retraso prom. por lw} + \text{retraso prom. por sw}$$

$$\text{retraso prom. fetch instrucciones: } 0,02 * 2(1+50+1) = 2,08$$

En los el cálculo anterior se utiliza la tasa de fallos promedio para instrucciones. Además que, por cada palabra enviada se tardan 2 ciclos en pasar por el bus su dirección y la palabra, y se tarda 50 ciclos resolviendo el fallo, como los bloques son de dos palabras entonces se multiplica por dos ese tiempo.

Luego, como la caché de datos es write back y write allocate, entonces los lw y los sw hacen exactamente lo mismo, de manera que no hay una diferenciación entre ambos, además que un 20% son bloques modificados, por lo que no todos se van a tener que escribir a memoria pues ya están en la caché, por lo que el retraso promedio por lw y sw viene dado por:

$$\text{retraso prom. por lw y sw: } 0.01 * [0.20 * (2*52 + 2*52) + 0.80 * (2*52)] = 1,248$$

Entonces el cálculo de CPI sería:

$$\text{CPI} = 6 + 2,08 + (0,3 + 0,1) * 1,248$$

$$\text{CPI} = 8,5792$$

R/ El CPI total es de 8,5792.

## Pregunta 4

- a. Se envía de memoria el bloque 8 a la caché, ocupando la posición 0 pues es mapeo directo. El bloque que se encuentra ahí está modificado, por lo que se envía al buffer víctima antes de que le caigan encima. Se lee de la p2 un -1.

Caché Datos Núcleo 1				
	0	1	2	3
p0	<del>-5</del> 25	25	9	5
p1	<del>-8</del> 26	26	10	6
p2	<del>3</del> -1	27	11	7
p3	4 -2	28	12	8
Etig.	36 8	5	22	11
Estado	<del>M</del> C	C	C	I

  

Buffer víctima Núcleo 1		
p0	-5	25
p1	-8	26
p2	3	-1
p3	4	-2
Etig.	36	8

- b. Ambas cachés tienen como compartido el bloque 5, entonces pueden leer sin problemas la palabra 0 con el valor 25.
- c. El núcleo 0 manda a invalidar el blk 22 en la caché del núcleo 1, con esto se escribe un 88 en la p0 del blk 22 en la caché del núcleo 0 y se pone modificado. Ahora el núcleo 1 pide escribir en el blk 22, por lo que se invalida en la caché del núcleo 0, se manda a su buffer víctima y se le envía a la caché del núcleo 1, la cual escribe un 45 en la p0 y deja el bloque como modificado.

Caché Datos Núcleo 0				
	0	1	2	3
p0	-7	25	9 88	15
p1	-8	26	10	5
p2	-9	27	11	17
p3	-10	28	12	20
Etig.	4	5	22	3
Estado	<del>M</del>	C	<del>C</del> M I	I

  

Buffer víctima Núcleo 0		
p0	5	88
p1	6	10
p2	9	11
p3	50	12
Etig.	11	22

 | Caché Datos Núcleo 1 |    |    |                  |    | |----------------------|----|----|------------------|----| |                      | 0  | 1  | 2                | 3  | | p0                   | 25 | 25 | 9 88 45          | 5  | | p1                   | 26 | 26 | 10               | 6  | | p2                   | -1 | 27 | 11               | 7  | | p3                   | -2 | 28 | 12               | 8  | | Etig.                | 8  | 5  | 22               | 11 | | Estado               | C  | C  | <del>C</del> I M | I  |     | Buffer víctima Núcleo 1 |    |    | |-------------------------|----|----| | p0                      | -5 | 25 | | p1                      | -8 | 26 | | p2                      | 3  | -1 | | p3                      | 4  | -2 | | Etig.                   | 36 | 8  | |

Pregunta 5

a.

PROCESADOR 0							PROCESADOR 1						
DIRECTORIO P0					CACHE P0		DIRECTORIO P1					CACHE P1	
BI	E	P0	P1		7 8	5 3	BI	E	P0	P1		9 5	5 8 3
0	M U		1		8 2	5 0	3	U				9 1	0 1 5 8
1	U			ET.	0 4	2 5	4	U M U	1		ET.	0 2	2 4 1
2	U C U			EST.	1 M	1 M C	5	C M C	1		EST.	M C	C M C
	C M U		1 1										
MP PROC 0 - PARTE COMPARTIDA							MP PROC 1 - PARTE COMPARTIDA						
BI 0		BI 1		BI 2			BI 3		BI 4		BI 5		
0	4	8	12	16	20	Dir	24	28	32	36	40	44	Dir
7 9	8 9	6	8	5	1 0 1	Valor	17	-6	2 8	2 5	3 5	4	Valor

b. En el hilillo 1:

- x1 = 1
- x2 = 0
- x3 = 5
- x4 = 5