

Lista 02

Questões Teóricas

① a) Programas que usam chamadas internas de hardware e que fazem uso de chamadas de sistema

$$b) \begin{array}{ccc} \text{sistema} & & \text{VM} \\ 10\% & \text{---} & 60\% \\ 20\% & \text{---} & x \end{array} \quad x = 120\%$$

$$c) \sum \text{Puros} = \frac{10\,452}{9} = 1161,33...$$

$$\sum \text{Puros} = \frac{3\,122,51}{9} = 346,9455...$$

d) Terism Null Call e Null I/O. Provavelmente, o motivo é que não se utiliza recursos complexos de E/S

② Ela precisa utilizar as funcionalidades de virtualização fornecidas pelo hardware e pela hipervisão para deter acesso ao hardware subjacente, gerenciar os modos de execução, interagir com o hipervisão e virtualizar seus próprios recursos para garantir o funcionamento adequado.

$$③ a) \text{Cada L1 Instrução} = \frac{32 \text{ KB}}{264 \text{ B} \cdot 4 \text{ bytes}} = \frac{2^{10}}{8} = \frac{2^{10}}{2^3} = 2^7 \text{ Conjunções}$$

$$\text{Cada L1 Dado} = \frac{32 \text{ KB}}{64 \text{ B} \cdot 8} = \frac{2^{10}}{2^4} = 2^6 \text{ Conjunções}$$

$$L2 = \frac{256 \text{ KB}}{64 \text{ B} \cdot 8} = \frac{2^8 \cdot 2^{10}}{2^5 \cdot 2^3} = \frac{2^{18}}{2^8} = 2^{10} \text{ Conjunções}$$

$$L3 = \frac{2 \text{ MB}}{64 \text{ B} \cdot 16} = \frac{2^7 \cdot 2^{10}}{2^6 \cdot 2^4} = \frac{2^{17}}{2^{10}} = 2^7 \text{ Conjunções}$$

b) Quanto a associatividade é tolerada para cache miss do cache, significa que o número de blocos de memória é multiplicado por 2. Com uma associatividade maior, há uma probabilidade maior de encontrar um bloco solicitado naquela cache.

c) Quanto mais dados na hierarquia de memória, mais tempo de acesso à memória, e consequentemente mais tempo demora para buscar e retornar o dado.

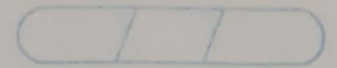
d) Bando - LRU; O bloco que foi usado por último no conjunto é substituído.

Bando - LRU com residências: Os blocos são reservados para sempre o acesso à cache.

④	TLB	Page Table	Cache	Possível? Se for, em que circunstâncias?
hit	hit	miss	Sim, o dado não está na cache mas está na memória principal	
miss	hit	hit	Sim, o TLB tem pouco espaço de armazenamento	
miss	hit	miss	Sim, o dado está na memória principal	
miss	miss	miss	Sim, o dado está no disco / Memória secundária	
hit	miss	hit/miss	Não	

	set	Address	set	Address		
⑤	0	16448	8		2 - miss	11 - hit
	1		9		3 - miss	3 - miss
	2	2	10		11 - miss	22 - miss
	3	103	11	11311	16 - miss	4 - miss
	4	4	12		21 - miss	27 - miss
	5	21	13	13	13 - miss	6 - miss
	6	26	14		64 - miss	11 - miss
	7		15		48 - miss	
					19 - miss	

LRU



b)	Cache Hit	Tag	Address	2 - miss	12 - miss
	0	[0, 1, 2, 3]	16, 64, 48, 4	3 - miss	11 - hit
	1	[0, 1, 2, 3]	21, 13,	11 - miss	3 - hit
	2	[0, 1, 2, 3]	2, 22, 6	16 - miss	22 - miss
	3	[0, 1, 2, 3]	3, 11, 12, 27	21 - miss	4 - miss
				13 - miss	27 - miss
				64 - miss	6 - miss
				48 - miss	11 - hit

⑥ $512 \times 64B = 32KB$

b) 16 palavras de 4B: 8 palavras $\times 8$ 8 = 64 palavras

memória A $\rightarrow 8 \text{ miss} + 56 \text{ hit}$

memória B $\rightarrow 64 \text{ miss} \times 8 = 512 \text{ miss}$

c) A memória A apresenta localidade espacial
A memória B não apresenta nenhuma

d) A computador é mais rápido porque a memória B acessa
por linha, apresentando a localidade espacial, reduzindo a taxa de miss.

⑦ Desmembramento direto:

Indice	Tag	Dado	
0			
1			
2			
...			
16k-2			
16k-1			

$16k = 2^{14}$

14b de índice

$22 - 14 - 2 = 16b \text{ de tag}$

Associação 2

$$\frac{16 \text{ KB}}{2} = 8 \text{ KB} = 2^{13}$$

index	valid	tag	Data	tag	Data
0					
1					
...					
8k-2					
8k-1					

13 b de index

$$32 - 13 - 2 = 17 \text{ b de tag}$$

Associação 4

index	valid	tag	Data	tag	Data	tag	Data	tag	Data
0									
1									
...									
4k-3									
4k-1									

$$\frac{16 \text{ KB}}{4} = 4 \text{ KB} = 2^{12}$$

12 b de index e 18 b de dados

Totalmente Associativa

index	valid	tag	Data	...	tag	Data
0				...		

$$32 - 2 = 30 \text{ b de tag}$$

1) A CPU acessa a TLB com o endereço virtual.

2.a) Se der hit, o dado é buscado na memória principal pelo endereço físico.

2.b) Se der miss, busca o endereço físico na Page Table.

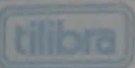
3.a) O endereço é atualizado na TLB e o dado é buscado na memória.

3.b) Se der miss na page table, o SO é informado como falta de página.

3.c) Então o dado é buscado na memória secundária.

4) O dado é trazido para a RAM, liberando uma página.

5) A Page Table é atualizada com o endereço da nova página na RAM.



9) a) Caches L1, L2 e L3

b) L1 → cache

L2 → 2 way

L3 → 4 way

c) 320 KB - L1

5 MB - L2

8 MB - L3

d) 256 B per core 3

e) 256 B virtual

512 B física

f) Get-ComputerInfo -ClassName Win32_CacheMemory -Property *