

- d) Um sistema de memória usa endereços de 32 bits. Ele possui uma cache com 256 blocos. A cache é totalmente associativa. Cada bloco contém 16 palavras. Cada palavra contém 4 bytes. Quantos bits tem o rótulo (TAG) a ser armazenado? **Justifique e mostre os cálculos mais relevantes.**

e) Afirmção: “Um controlador de DMA aciona a linha de interrupção do barramento cada vez que um novo item precisar ser transferido de um dispositivo de E/S para a memória”. A afirmação é verdadeira ou falsa ? **Justifique.**

f) Um sistema contém apenas um nível de cache. Os tempos de acesso à memória principal e à cache são 30ns e 3ns, respectivamente. Determine a taxa de fracassos **mr** (miss rate) para se obter um tempo de acesso médio de 3,5 ns e mostre seus cálculos.

**Resposta: mr =**

**Cálculos:**

## **Parte II [valor: 5,0 pontos]**

2. [2,0: 4 x 0,5] Considere três processadores com diferentes configurações de cache:

- Cache 1: Mapeamento direto com 1 palavra por bloco;
- Cache 2: Associativa com 2 alternativas por conjunto (2-way) com 4 palavras por bloco.

Foram feitas as seguintes medidas de taxas de fracasso (miss) de instruções (imr) e de dados (dmr):

- Cache 1: imr = 4%; dmr = 6%;
- Cache 2: imr = 2%; dmr = 3%.

Para esses processadores, metade das instruções contém uma referência a dados. Assuma que a penalidade de falta é  $6+T$ , onde  $T$  é o tamanho do bloco expresso em palavras. Sendo  $I$  o número de instruções executadas no programa, determine os valores os abaixo, mostrando o resultado final e os cálculos intermediários.

a) O número médio de **ciclos de parada** por instrução devido a faltas na Cache 1.

**Resultado: CPI<sub>stall</sub> (Cache 1) =**

**Cálculos: ciclos =**

b) O número médio de **ciclos de parada** por instrução devido a faltas na Cache 2.

**Resultado: CPI<sub>stall</sub> (Cache 2) =**

**Cálculos: ciclos =**

Para a carga de trabalho utilizada no experimento, mediu-se um CPI total igual a 2 ao se usar o processador junto com a Cache 1. Calcule:

c) O número total médio de **ciclos por instrução** ao se usar a Cache 2.

**Resultado: CPI<sub>total</sub> (Cache 2) =**

**Cálculos:**

d) Supondo endereços de 32 bits e que a Cache 2 armazene 1024 blocos, calcule o número total de bits necessários para implementá-la, incluindo dados, rótulo (TAG) e bit de validade.

**Resultado: número de bits =**

**Cálculos:**

A Tabela 1 é usada como referência para resolver as Questões 3 e 4. Ela mostra de forma simbólica, para cada endereço de memória na faixa de 0x0 a 0x5C (coluna 1), o conteúdo da memória principal (coluna 3). A coluna 2 é um campo para auxiliar você na correspondência entre endereços hexadecimais e binários.

**Tabela 1** – Conteúdo (parcial) da memória principal

Endereço (0x)	End. [7:0] (0b)	Conteúdo
0000 0000		A
0000 0004		B
0000 0008		C
0000 000C		D
0000 0010		K
0000 0014		L
0000 0018		M
0000 001C		N
0000 0020		E
0000 0024		F
0000 0028		G
0000 002C		H
0000 0040		W
0000 0044		X
0000 0048		Y
0000 004C		Z
0000 0050		P
0000 0054		Q
0000 0058		R
0000 005C		S

**Tabela 2** – Status da cache após cada acesso

Bloco	00				01				10				11			
Palavra	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
Acesso 1																
Acesso 2																
Acesso 3																
Acesso 4																
Acesso 5																

**Tabela 3** – Status da cache após cada acesso

ac	Bloco 0				ct	Bloco 1				ct	Bloco 2				ct	Bloco 3				ct	F/S
	00	01	10	11		00	01	10	11		00	01	10	11		00	01	10	11		
1																					
2																					
3																					
4																					
5																					
6																					
7																					

3. [5 x 0,2 = 1,0] Seja a seguinte sequência de referências à memória: 0x04, 0x20, 0x10, 0x14, 0x58. Considere uma cache com mapeamento direto, inicialmente vazia, cujo tamanho total é de 16 palavras, sendo que cada bloco contém 4 palavras. Preencha a Tabela 2 de acordo com a seguinte orientação: cada linha da Tabela 2 representa o estado de cada um dos blocos da cache após os acessos 1, 2, ... , 5 induzidos pela sequência de endereços dada; indique o conteúdo de cada bloco válido, explicitando todas as suas palavras.

4. [4 x 0,5 = 2,0] Seja a seguinte sequência de referências à memória: 0x08, 0x18, 0x24, 0x1C, 0x54, 0x48, 0x40. Suponha uma cache totalmente associativa, inicialmente vazia, com 4 palavras por bloco e um total de 16 palavras. Assuma que a substituição de elementos siga o critério LRU. Mostre o conteúdo da cache imediatamente após cada um dos acessos induzidos pela sequência de endereços dada. Para isso, preencha a Tabela 3 de acordo com a seguinte orientação: Cada linha da Tabela 3 representa o estado de cada um dos blocos da cache num dado instante de tempo designado pelo número do acesso à memória, “ac”. Indique o conteúdo de cada bloco válido, explicitando todas as suas palavras. Suponha que, havendo blocos livres na cache, os blocos trazidos da memória sejam armazenados no bloco livre de menor número (o preenchimento dos blocos na Tabela 3 deve ser da esquerda para a direita). O contador “ct” é um campo para auxiliar você a monitorar o critério LRU. Atualize o contador no campo “ct” com o número do acesso em que o bloco foi referenciado pela última vez. Anote se ocorreu sucesso (hit) ou fracasso (mis) na coluna “F/S”.

### **Parte III [valor: 2,0 pontos]**

5. [3 x 0,33.. = 1,0] Em um sistema com apenas um nível de cache, durante a execução de uma instrução *load*, pode ocorrer um acesso de *escrita* na memória principal, dependendo do mecanismo utilizado para manter a consistência entre cache e memória principal. Descreva as condições necessárias e suficientes para que esse evento ocorra:

**Condição 1:**

**Condição 2:**

**Condição 3:**

6. [2 x (0,25+0,25) = 1,0] Segundo a análise clássica de algoritmos, no pior caso, o tempo de execução do algoritmo QuickSort tem um limite superior assintótico dado por uma função linear, o que é denotado por  $O(n)$ , enquanto o algoritmo RadixSort é limitado por uma função quadrática, o que é denotado por  $O(n^2)$ , onde  $n$  é o número de itens a ordenar. (Lembrete: o comportamento assintótico refere-se à tendência observada no tempo de execução para um número grande de itens a ordenar). As curvas das Figuras A e B do Anexo I representam, respectivamente, o número de instruções executadas por item e o número de ciclos de relógio gastos por item, como função do número total de itens a ordenar.
- a) Qual das figuras (A ou B) correlata melhor com a previsão da análise clássica?
    - Justifique argumentando com a tendência assintótica das curvas nela representadas.
    - Aponte que fatores determinantes do tempo de execução (I, CPI, f) não são capturados nessa figura.

**Figura:**

**Justificativa:**

**Apontamento:**

- b) Qual das figuras (A ou B) diverge da previsão da análise clássica ?
  - Justifique invocando fenômeno(s) não levado(s) em conta pela análise clássica de algoritmos.
  - Aponte que fatores determinantes do tempo de execução (I, CPI, f) não são capturados nessa figura.

**Figura:**

**Justificativa:**

**Apontamento:**

## ANEXO I

