

Sistemas Operacionais

Gabarito de lista de Exercícios Memória / Entrada e Saída

1. Um controlador DMA tem 4 canais. O controlador é capaz de requisitar uma palavra de 32 bits a cada 100ns. Uma resposta leva o mesmo tempo. Até que ponto o barramento deve ser rápido para evitar a ocorrência de um gargalo?

Cada barramento permite uma solicitação e uma resposta, cada um levando 100ns, ou 200ns por transação no barramento. Isto dá 5 milhões de transações por barramento/sec. Se cada um é bom para 4bytes, o barramento tem que permitir 20 MB/sec. O fato dessa transação poder ser efetuada em 4 barramentos por round-robin por exemplo é irrelevante. Uma transação no barramento leva 200ns, independentemente se são efetuadas para o mesmo dispositivo ou para dispositivos diferentes, então o numero de canais de DMA também não importa.

2. Suponha que um computador possa ler ou escrever uma palavra de memória em 10 ns. Suponha também que, quando uma interrupção ocorre, todos os 32 registradores da CPU mais o contador de programa e a PSW são colocados na pilha. Qual é o número máximo de interrupções por segundo que essa máquina pode processar?

São 34 blocos de memória que serão empilhados e posteriormente desempilhados durante uma interrupção. Desconsiderando o tempo de execução da própria interrupção, temos ao todo 68 operações de leitura e escrita na memória:

```
tempo_int = 2 * (32 + 1 + 1) * 10 = 680 ns = 680*10-9 segundos
qtde_int_max = 1 : 680*10-9 = (1/680)^109
qtde_int_max < 1.470.588,23529
```

3. Uma típica página de texto impressa contém 50 linhas de 80 caracteres cada. Imagine que uma certa impressora possa imprimir seis páginas por minuto e que o tempo para escrever um caractere no registrador de saída da impressora é tão pequeno que possa ser ignorado. Tem sentido usar essa impressora com E/S orientada à interrupção se cada caractere impresso requer uma interrupção que leva 50 ms para ser servida?

Para começar, encontraremos as seguintes variáveis obtidas do texto facilmente:

```
char_pag = 50 x 80 = 4.000 caracteres por pagina
char_min = 6 x char_pag = 6 x 4.000 = 24.000 caracteres por minuto
char_seg = char_min / 60 = 400 caracteres por segundo
t_char = 2,5 ms
```

A frequência de E/S orientada a interrupções é calculada abaixo: ($f = 1 / t$)

a.) para $t = 50$ ms
 $qtde_mili = 1 / 50$ ms = 20 caracteres

b.) para $t = 50$ μ s
 $qtde_micro = 1 / 50$ μ s = 20.000 caracteres

Cada caractere usa 50 do tempo de CPU para a interrupção, então, coletivamente em cada segundo a sobrecarga da interrupção é de 20 msec. Usando interrupção (I/O), o tempo restante 980 ms estarão disponíveis para outro job. (a interrupção custa apenas 2% deixando o processador livre o restante do tempo)

4. O que é 'independência de dispositivo'?

É a camada de software, usualmente uma interface uniforme (API), que permite que os processos a nível de usuário se comuniquem com os dispositivos sem depender de como estes estão implementados.

Ex: A syscall fictícia escreva(frase, devNo) pode enviar uma mensagem escrita tanto para o monitor, utilizando um devNo padrão, como poderia enviar para a impressora.

Os drivers de dispositivo são elementos importantes neste processo de independência.

5. Em qual das 4 camadas de software de E/S se realiza cada uma das seguintes atividades?

- a) Calcular trilha x setor x cabeçote para uma leitura em disco;
- b) Escrever comandos no registrador do dispositivo;
- c) Verificar se o usuário tem permissão de utilizar o dispositivo;
- d) Converter inteiros binários em ASCII para a impressão.

a.) Driver de dispositivo.

b.) Apenas o driver de dispositivo conhece os endereços dos registradores e valores que devem ser utilizados para que o hardware funcione corretamente.

c.) Software independente de dispositivo.

d.) O nível de processos de usuários é responsável pelas chamadas de E/S e também pela formatação da informação que será entregue posteriormente aos drivers

6. Considere um sistema de memória virtual paginada cujos endereços virtuais possuem 30 bits. A tabela de páginas desse sistema possui 1M linhas. A memória RAM do computador onde esse sistema opera possui 16M palavras. O Sistema Operacional é carregado no topo da memória RAM, ocupando 500K palavras. Responda:

6.1) Qual o tamanho das páginas virtuais (PV) e qual o tamanho das páginas reais (PR)?

- a) PV=2 Kpalavras e PR=1 Kpalavras.
- b) PV = PR = 2 Kpalavras.
- c) PV = PR = 1 Kpalavras.**
- d) PV = PR = 0,5 Kpalavras.
- e) PV = PR = 1 Mpalavras.

6.2) Qual o tamanho máximo de um processo para que ele possa ser carregado totalmente na memória real?

- a) 16 Mpalavras.
- b) 1 Gpalavras.
- c) 1 Kpalavras.
- d) 2,5 Kpalavras.
- e) n.d.a.**

6.3) Se o espaço utilizado para o sistema operacional ficar fixo (o sistema operacional não sofre paginação), qual o primeiro endereço real que pode ser utilizado por um processo, considerando-se que o sistema operacional ocupa as páginas de maior numeração (topo da memória).

- a) Zero.**
- b) 15.5M.
- c) 15M.
- d) 16M.
- e) 500K.

6.4) Se o endereço virtual de número 10480 estiver mapeado no endereço real de número 51440, onde estará mapeado o endereço virtual de número 11000?

- a) Não é possível calcular porque os endereços virtuais 10480 e 11000 não pertencem à mesma página.

- b) Endereço virtual 11000 estará mapeado na posição real 51960.
- c) Não é possível calcular porque o endereço virtual de número 10480 nunca poderá ser mapeado no endereço real 51440.
- d) Endereço virtual 10480 não poderia estar mapeado na posição 51440 porque essa posição estará ocupada pelo sistema operacional.
- e) n.d.a.

6.5) Se o disco do sistema computacional considerado tiver capacidade para armazenar 4.3 GBytes, qual o espaço que estará disponível para o sistema de arquivos?

- a) 4.3 Gbytes.
- b) 2.3 Gbytes.
- c) 3.3 Gbytes.
- d) 1 Gbytes.
- e) n.d.a.

6.6) Se o endereço virtual tem 30 bits então se pode afirmar que:

- a) O processador utilizado no sistema pode manipular endereços com no máximo 30 bits.
- b) O processador utilizado no sistema pode manipular endereços com no mínimo 30 bits.
- c) Se o processador utilizado no sistema manipular endereços com até 20 bits a memória virtual continuará sendo de 1 Gbytes.
- d) O gerenciador de memória utilizado considera 30 bits no endereço e o processador utilizado pode ser capaz de manipular endereços com mais de 30 bits.
- e) n.d.a.

7. Explique o funcionamento da técnica de DMA e sua principal vantagem.

Olhar os slides, especialmente os específicos de DMA.

8. Em um sistema paginado, as páginas têm 4Kb endereços, a memória principal possui 32Kb e o limite de páginas na memória principal é de 8 páginas. Um processo faz referência aos endereços virtuais situados nas páginas 0, 2, 1, 9, 11, 4, 5, 2, 3, 1, nessa ordem. Após essa seqüência de acessos, a tabela de páginas desse processo tem a configuração abaixo, sendo que as entradas em branco correspondem às páginas ausentes.

Página Virtual	Endereço Real
0	8 kB
1	4 kB
2	24 kB
3	0 kB
4	16 kB
5	12 kB
6	*
7	*
8	*
9	20 kB
10	*
11	28 kB
12	*
13	*
14	*
15	*

- a) Qual o tamanho (em bits) e o formato do endereço virtual? Justifique sua resposta.

4 bits para página + 12 bits para deslocamento = 16 bits por endereço virtual

- b) O processo faz novas referências aos endereços virtuais situados nas páginas 5, 15, 12, 8, 0, 15, 6, 10, 11 e 13, nessa ordem. Simule o processamento dessa sequência de acessos utilizando o algoritmo de troca de páginas FIFO e calcule as faltas de páginas.

FIFO – 5 15 12 8 0 15 6 10 11 13

0	8k	15 *	15	15	15	15	15	15	15	0k = 13
2	24k	2	12 *	12	12	12	12	12	12	4k = 8
1	4k	1	1	8 *	8	8	8	8	8	8k = 15
9	20k	9	9	9	0 *	0	0	0	0	12k = 11
11	28k	11	11	11	11	6 *	6	6	6	16k = 10
4	16k	4	4	4	4	4	10 *	10	10	20k = 0
5	12k	5	5	5	5	5	5	11 *	11	24k = 12
3	0k	3	3	3	3	3	3	3	13 *	28k = 6

Total de faltas de páginas = 8