



Curso de Tecnologia em Sistemas de Computação  
Disciplina de Sistemas Operacionais  
**Professores:** Valmir C. Barbosa e Felipe M. G. França  
**Assistente:** Alexandre H. L. Porto

Quarto Período  
AP2 - Segundo Semestre de 2011

Nome -

Assinatura -

---

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
  2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  3. Você pode usar lápis para responder as questões.
  4. Ao final da prova devolva as folhas de questões e as de respostas.
  5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. (2,5) Diga se as seguintes afirmativas são falsas ou verdadeiras. Para responder, escreva apenas F ou V para cada item em seu caderno de respostas.

- (a) (0,5) Se o hardware do computador não tiver uma DMA, então o processador será o responsável pela transferência de dados entre a controladora de um dispositivo físico e a memória.

**Resp.:** V (Verdadeira).

- (b) (0,5) O conceito de independência de dispositivo define que as operações executadas em um dispositivo físico independem do seu tipo, ou seja, o mesmo conjunto de operações pode ser usado para acessar todos os dispositivos físicos do computador.

**Resp.:** F (Falsa, porque a independência de dispositivo define que as operações executadas independem do funcionamento interno dos dispositivos físicos de um mesmo tipo como, por exemplo, as diversas marcas de discos rígidos).

- (c) (0,5) Um recurso preemptivo pode ser tirado de um processo sem prejudicar o resultado do que está sendo computado pelo processo. Já um recurso não-preemptivo não pode ser tirado de um processo, porque isso prejudicaria o resultado que o processo está computando.

**Resp.:** V (Verdadeira).

- (d) (0,5) A condição de espera circular ocorre quando um processo é bloqueado ao esperar por dados a serem disponibilizados em um buffer circular usado por uma controladora de um dispositivo físico.

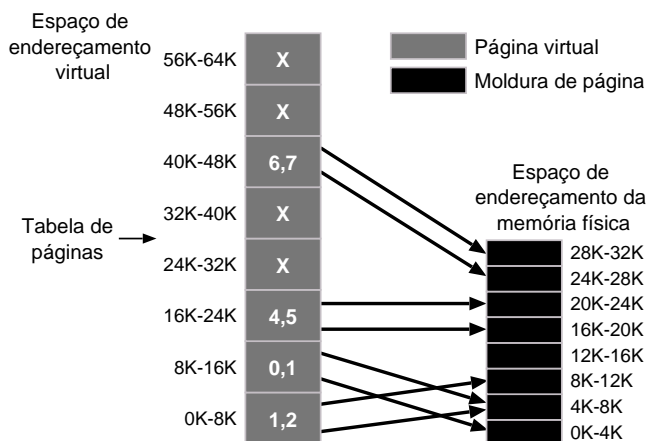
**Resp.:** F (Falsa, pois a condição de espera circular ocorre quando existe uma cadeia circular de dependências entre um conjunto de processos que estão bloqueados esperando por recursos pertencen-

tes a outros processos do mesmo conjunto).

- (e) (0,5) Considere a técnica para evitar impasses, vista na Aula 7, que numera cada recurso com um número inteiro diferente, e na qual todo processo é obrigado a solicitar os recursos de que precisa em ordem estritamente crescente desses números. Essa técnica continua a ser válida se dois ou mais recursos têm o mesmo número inteiro associado a eles.

**Resp.:** V (Verdadeira).

2. (1,5) Um aluno de sistemas operacionais disse que a figura a seguir representa um mapeamento correto entre páginas virtuais e molduras de página. Se você acha que a figura do aluno está correta, basta dizer que sim mas, se você acha que existe algo de errado com a figura, então indique cada um dos erros que você acha que o aluno cometeu.



**Resp.:** A figura do aluno está errada. O primeiro erro é que as molduras de página não têm o mesmo tamanho das páginas virtuais, já que cada moldura tem 4K de tamanho e cada página tem 8K de tamanho. O segundo erro, decorrente do primeiro, é que cada página virtual está mapeada em duas molduras de página. Como vimos na Aula 8, uma página virtual deve sempre estar mapeada em uma única moldura de página. Finalmente, o terceiro erro é que a moldura 1 está simultaneamente armazenando os 4K iniciais da página 0 e os 4K finais da página

1 e, como notamos antes, somente uma página virtual pode estar mapeada em uma moldura.

3. (1,5) Considere a execução do algoritmo LRU dada a seguir, que mostra como as páginas virtuais são substituídas conforme um processo acessa, de cima para baixo, as páginas dadas na tabela. Suponha que o sistema operacional passe a usar o algoritmo de segunda chance. Suponha ainda que o bit referenciada de uma página seja ligado quando ela é copiada para a memória e desligado se duas outras páginas são acessadas antes de ela ser novamente acessada. A execução do novo algoritmo, supondo que o processo novamente acesse as páginas da tabela na mesma ordem, causa a mesma sequência de substituições? Justifique a sua resposta.

Página	Página a ser substituída se necessário
0	0
2	0
3	0
2	0
1	3
2	3
3	1
7	2

**Resp.:** Primeiramente precisamos descobrir o número de molduras de página alocadas ao processo. Note, pela tabela que quando o processo acessa a página 1 a página virtual a ser substituída muda de 0 para 3. Como a página 0 foi acessada somente quando foi copiada para a memória, então podemos concluir que ela foi substituída ao ser acessada a página 1. Agora, como nenhuma outra substituição ocorreu antes do acesso à página 1, porque a página 0, antes desse acesso, era sempre a página a ser substituída, então podemos concluir que o número de molduras alocadas ao processo é igual ao número de páginas diferentes acessadas antes da página 1. Como as páginas 0, 2 e 3 foram as únicas acessadas antes da página 1, então o número de molduras alocadas ao processo é 3. Ao usarmos o algoritmo de segunda chance, onde as

páginas a serem substituídas são ordenadas, em ordem crescente, de acordo com o tempo em que elas foram copiadas na memória, temos as escolhas dadas na tabela a seguir. Note que, nesse algoritmo, o bit **referenciada** da página copiada há mais tempo na memória se tornará 0 se ele for igual a 1, pois esta página deixará de ser escolhida e será colocada no final da ordenação. Nesta tabela, na primeira coluna, damos a página acessada, sendo que as páginas são mostradas na mesma ordem dada na tabela do enunciado. Na segunda coluna mostramos a ordem em que as páginas foram copiadas na memória, sendo que ao lado de cada página damos, entre parênteses, o valor atual do seu bit **referenciada**. Nessa coluna também destacamos, em negrito, as páginas que são as substituídas. Como dito no enunciado, o bit referenciada de uma página é limpo se, depois de ser acessada a página, duas páginas diferentes dela forem acessadas. Pela tabela, vemos que a sequência de substituições é a mesma do que a do algoritmo LRU.

Página	Ordem de escolha		
0	<b>0(1)</b>		
2	<b>0(1)</b>	2(1)	
3	<b>0(0)</b>	2(1)	3(1)
2	<b>0(0)</b>	2(1)	3(1)
1	2(1)	<b>3(0)</b>	1(1)
2	2(1)	<b>3(0)</b>	1(1)
3	2(1)	3(1)	<b>1(0)</b>
7	<b>2(0)</b>	3(1)	7(1)

4. (1,5) Suponha que o sistema operacional use a segmentação com paginação. O que ocorrerá se existir somente um segmento compartilhado por todos os processos em execução?

**Resp.:** Como existe somente um segmento compartilhado por todos os processos, então o espaço de endereçamento definido pelos endereços que podem ser acessados dentro desse segmento será equivalente a um espaço de endereçamento virtual da paginação. A diferença é que agora teremos um único espaço de endereçamento (e uma única tabela de páginas) para todos os processos, ao invés de um espaço de

endereçamento virtual para cada processo. Uma outra diferença é que o tamanho desse espaço será variável e dependerá do tamanho do segmento. Finalmente, o método de gerenciamento não irá mais garantir a segurança dos dados dos processos.

5. (1,5) Qual é a principal inovação introduzida pelo acesso aleatório aos blocos de um arquivo em relação ao acesso sequencial?

**Resp.:** A principal inovação é a operação **seek**, que permite acessar diretamente qualquer bloco do arquivo de modo independente do último bloco acessado. Essa operação, em essência, permite acessar um bloco posterior ao último bloco acessado sem precisar acessar os blocos entre esses dois blocos, ou um bloco anterior ao último bloco acessado sem precisar executar a operação **rewind** e, depois disso, acessar todos os blocos desde o inicial até o bloco que se deseja acessar.

6. (1,5) Suponha que o sistema operacional use uma lista encadeada para gerenciar os blocos livres do disco. O que ocorrerá se o endereço de um bloco livre estiver armazenado em duas ou mais entradas da lista? E se o endereço de um bloco livre não estiver armazenado em nenhuma entrada da lista?

**Resp.:** -Se o endereço de um bloco estiver armazenado em duas ou mais entradas da lista encadeada, então o bloco associado ao endereço poderá ser alocado a pelo menos dois arquivos diferentes. Logo, a integridade do conteúdo dos arquivos que incorretamente compartilham esse bloco não poderá mais ser garantida porque, quando o bloco for alterado pela escrita em um dos arquivos, todos os outros arquivos terão os seus conteúdos parcialmente alterados.

-Se o endereço de um bloco não estiver armazenado em nenhuma entrada da lista encadeada, e se esse bloco estiver alocado a algum arquivo, então não haverá nenhum problema. Porém, se o bloco não estiver alocado a um arquivo então o bloco, chamado de **bloco ausente**, apesar de estar livre, não poderá ser alocado a um arquivo até que ele seja armazenado na lista encadeada.