



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
Gabarito da AD2 - Segundo Semestre de 2012

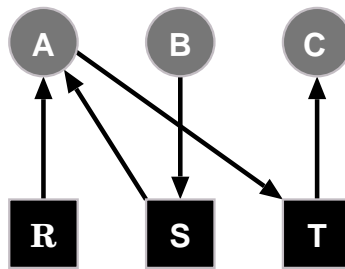
Atenção: ADs enviadas pelo correio devem ser postadas cinco dias antes da data final de entrega estabelecida no calendário de entrega de ADs.

Atenção: Cada aluno é responsável por redigir suas próprias respostas. Provas iguais umas às outras terão suas notas diminuídas. As diminuições nas notas ocorrerão em proporção à similaridade entre as respostas. Exemplo: Três alunos que respondam identicamente a uma mesma questão terão, cada um, 1/3 dos pontos daquela questão.

Observação: A questão 5 é opcional. Se você tentar resolvê-la, você poderá ganhar até 2,0 pontos adicionais na nota da prova, mas a sua nota ainda será limitada a 10,0 pontos.

Nome -
Assinatura -

1. (2,0) Responda as seguintes perguntas, considerando o grafo de alocação de recursos dado a seguir, e supondo que um processo é bloqueado ao tentar obter um recurso já alocado a outro processo:



- (a) (1,0) O que ocorreria se o único processo não-bloqueado tentasse obter um dos recursos que ele não possui, supondo que todos os recursos são não-preemptivos?

Resp.: Note que o processo B não possui recursos e foi bloqueado ao requisitar o recurso S, que o processo A possui os recursos R e S e foi bloqueado ao requisitar o recurso T, e que o processo C, que possui o recurso T, é o único processo não bloqueado. Como todos os recursos são não-preemptivos, ocorreria um impasse se C requisitasse R (ciclo C-R-A-T-C) ou S (ciclo C-S-A-T-C).

- (b) (1,0) O que ocorreria se o único processo não-bloqueado tentasse obter um dos recursos que ele não possui, supondo que o recurso em questão é compartilhado ao invés de dedicado (não-preemptivo)?

Resp.: Como B está bloqueado a espera de S, o enunciado necessariamente refere-se ao caso de o recurso R ser preemptivo, o que é necessário para ele ser compartilhado. Nesse caso, ao contrário do item (a), não ocorreria nenhum impasse, se C requisitasse R. Esse recurso estaria compartilhado por A e C.

2. (2,0) Considere um computador com uma memória virtual de 64KB, com páginas virtuais de 4KB de tamanho. Considere ainda que a memória física tem 25% do tamanho da memória virtual. Como será preenchida a tabela de páginas, se os endereços virtuais 22514, 65535, 32782 e 6070 estiverem mapeados, respectivamente, nos endereços físicos 10226, 8191, 14 e 14262? Na sua resposta, use o mesmo estilo usado

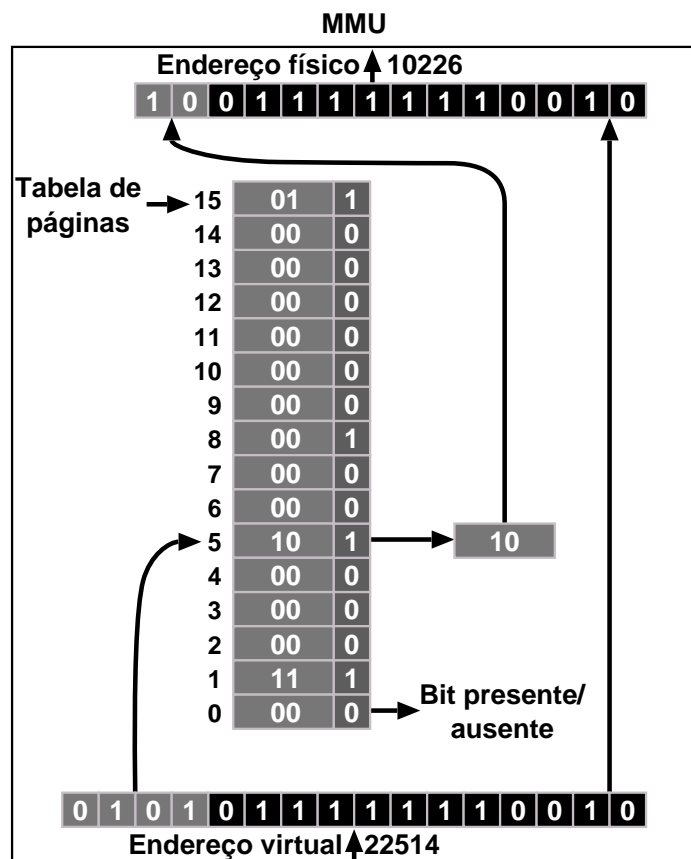
na transparência 20 da Aula 8. Além disso, suponha que, inicialmente, todas as molduras estão vazias.

Resp.: Pela questão, vemos que a memória virtual, com 64KB, requer endereços de 16 bits, e que ela é dividida em 16 páginas virtuais de 4KB. Assim, os 4 bits superiores do endereço virtual definem o número da página e os 12 bits inferiores o deslocamento dentro dessa página. Também pela questão, vemos que a memória física tem 16KB (25% ou 1/4 do tamanho da memória virtual), requerendo endereços de 14 bits. Como o tamanho da moldura de página é igual ao da página virtual, então a memória física é dividida em 4 molduras de página. Assim, os 2 bits superiores do endereço físico indicam o número da moldura e os 12 bits inferiores o deslocamento dentro dessa moldura. Logo, pelo mapeamento dado na questão, observamos o seguinte:

- O endereço virtual 22514, que em binário é 0101011111110010, estar mapeado no endereço físico 10226, que em binário é 1001111111-0010, significa que a página virtual 0101 em binário ou 5 em decimal está mapeada na moldura de página 10 em binário ou 2 em decimal.
- O endereço virtual 65535, que em binário é 1111111111111111, estar mapeado no endereço físico 8191, que em binário é 0111111111-1111, significa que a página virtual 1111 em binário ou 15 em decimal está mapeada na moldura de página 01 em binário ou 1 em decimal.
- O endereço virtual 32782, que em binário é 1000000000001110, estar mapeado no endereço físico 14, que em binário é 000000000011-10, significa que a página virtual 1000 em binário ou 8 em decimal está mapeada na moldura de página 00 em binário ou 0 em decimal.
- O endereço virtual 6070, que em binário é 0001011110110110, estar mapeado no endereço físico 14262, que em binário é 110111101-10110, significa que a página virtual 0001 em binário ou 1 em decimal está mapeada na moldura de página 11 em binário ou 3 em decimal.

Logo, a tabela de páginas é a dada na figura a seguir, lembrando que o bit presente/ausente é igual a 1 para as páginas virtuais mapeadas em

alguma moldura de página e 0 para as páginas não mapeadas. Nessa figura mostramos, como ilustração, a conversão do endereço virtual 22514 para o endereço físico 10226.



3. (3,0) Considere que um processo tenha acessado as páginas virtuais 0, 1, 3, 4, 3, 2, 1, 7, 9 e 0 na ordem dada. Considere também o algoritmo LRU, o algoritmo de segunda chance e o algoritmo NRU. No algoritmo de segunda chance e no NRU, o bit R de uma página é mudado para 1 quando ela é acessada e para 0 somente depois de duas páginas diferentes dela serem acessadas. Em todos os algoritmos, é verdade que a mesma página será substituída caso ocorra uma nova falha de página, supondo que somente as páginas 0, 7 e 9 foram modificadas sempre que

acessadas? Justifique a sua resposta.

Resp.: Como observado no aviso que foi publicado na plataforma, foram alocadas 4 molduras de página inicialmente vazias ao processo. Primeiramente vamos mostrar, na tabela a seguir, a sequência de acessos às páginas virtuais dadas na questão para o algoritmo LRU. Como vimos na aula 9, no algoritmo LRU, as páginas são primeiramente ordenadas, em ordem crescente, de acordo com o tempo do seu último acesso. A página a ser substituída é a primeira página segundo essa ordenação, isto é, a página não acessada há mais tempo. Na tabela dada a seguir mostramos, em cada linha, o que ocorre ao acessarmos as páginas na ordem dada no enunciado. Para cada uma dessas linhas mostramos na primeira coluna a página que é acessada e na segunda coluna a ordem em que as páginas devem ser escolhidas. Como podemos ver pela última linha da tabela, a página 1 será substituída caso uma nova falha de página ocorra.

| Páginas | Ordenação |
|---------|-----------|
| 0 | 0 |
| 1 | 0 1 |
| 3 | 0 1 3 |
| 4 | 0 1 3 4 |
| 3 | 0 1 4 3 |
| 2 | 1 4 3 2 |
| 1 | 4 3 2 1 |
| 7 | 3 2 1 7 |
| 9 | 2 1 7 9 |
| 0 | 1 7 9 0 |

Como também vimos na aula 9, ao usarmos o algoritmo de segunda chance, onde as páginas a serem escolhidas são ordenadas, em ordem crescente, de acordo com o tempo em que elas foram copiadas na memória, teremos 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 essa página deixará de ser escolhida e será colocada no final da ordenação. Na tabela

dada a seguir, na primeira coluna, damos a página acessada, sendo que as páginas são mostradas na mesma ordem dada no enunciado. Na segunda coluna mostramos a ordem em que as páginas são escolhidas pelo algoritmo, sendo que ao lado de cada página damos, entre parênteses, o valor atual do seu bit **referenciada**. Pela última linha da tabela, vemos que a página 1 também será a próxima a ser substituída.

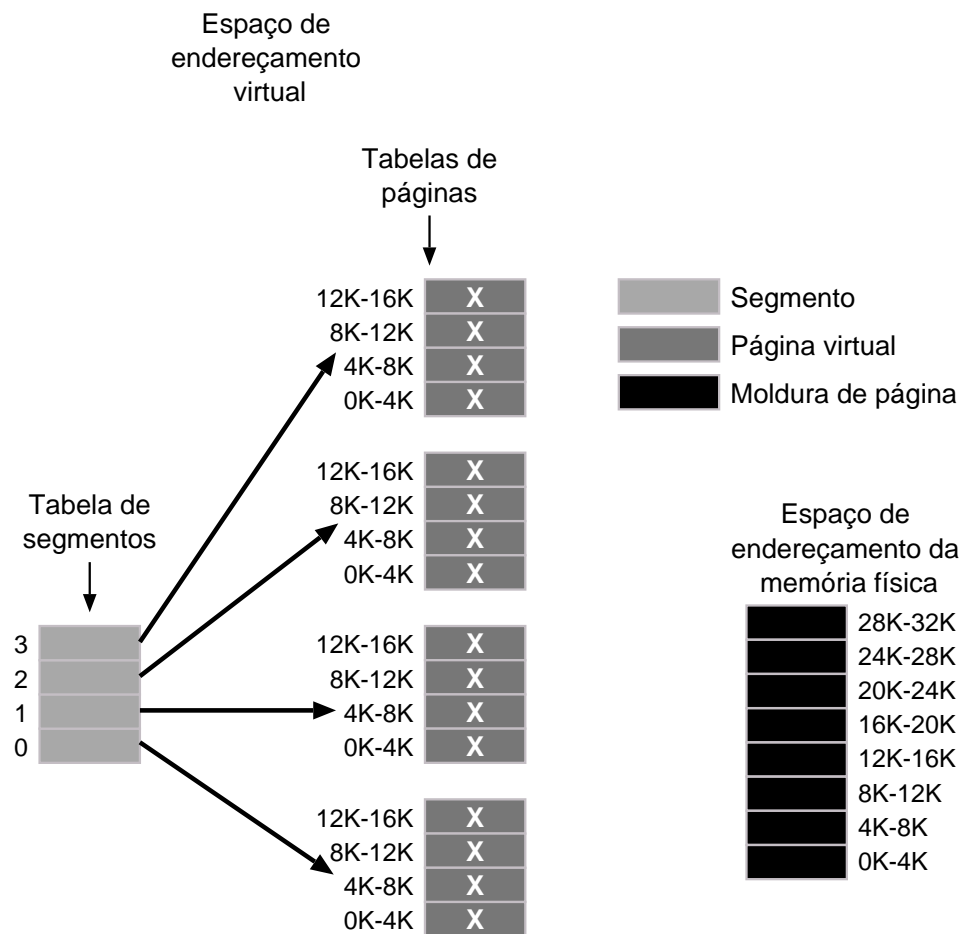
| Página | Ordem de escolha | | | | |
|--------|------------------|------|------|------|--|
| 0 | 0(1) | | | | |
| 1 | 0(1) | 1(1) | | | |
| 3 | 0(0) | 1(1) | 3(1) | | |
| 4 | 0(0) | 1(0) | 3(1) | 4(1) | |
| 3 | 0(0) | 1(0) | 3(1) | 4(1) | |
| 2 | 1(0) | 3(1) | 4(0) | 2(1) | |
| 1 | 1(1) | 3(0) | 4(0) | 2(1) | |
| 7 | 3(0) | 4(0) | 2(0) | 1(0) | |
| | 4(0) | 2(0) | 1(0) | 7(1) | |
| 9 | 2(0) | 1(0) | 7(1) | 9(1) | |
| 0 | 1(0) | 7(0) | 9(1) | 0(1) | |

Finalmente, como também vimos na aula 9, no algoritmo NRU, as páginas são primeiramente divididas em 4 classes diferentes, de acordo com o valor dos bits **modificada** (M) e **referenciada** (R): Classe 0: ambos os bits R e M são 0; Classe 1: o bit R é 0 e o bit M é 1; Classe 2: o bit R é 1 e o bit M é 0; e Classe 3: ambos os bits R e M são 1. Depois de dividir as páginas em classes, a página a ser substituída será uma das páginas da classe não vazia com o menor número. Como vimos na aula 9, o bit R sempre é ligado quando a página é acessada. Na tabela dada a seguir mostramos, em cada linha, como as páginas na memória são divididas em cada uma das 4 classes, após acessarmos a página dada na primeira coluna, de acordo com a ordem dada no enunciado. As classes são mostradas nas colunas de 2 até 5. Como podemos ver por essa tabela, a próxima página a ser substituída será também a 1. Note que somente as páginas 0, 7 e 9 podem estar nas classes 1 ou 3, pois somente para elas, que são alteradas, o bit M é 1.

| Página | Classes | | | |
|--------|---------|---|------|------|
| | 0 | 1 | 2 | 3 |
| 0 | - | - | - | 0 |
| 1 | - | - | 1 | 0 |
| 3 | - | 0 | 1, 3 | - |
| 4 | 1 | 0 | 3, 4 | - |
| 3 | 1 | 0 | 3, 4 | - |
| 2 | 4 | 0 | 2, 3 | - |
| 1 | 3 | 0 | 1, 2 | - |
| 7 | 2 | 0 | 1 | 7 |
| 9 | 1 | 0 | - | 7, 9 |
| 0 | 1 | 7 | - | 0, 9 |

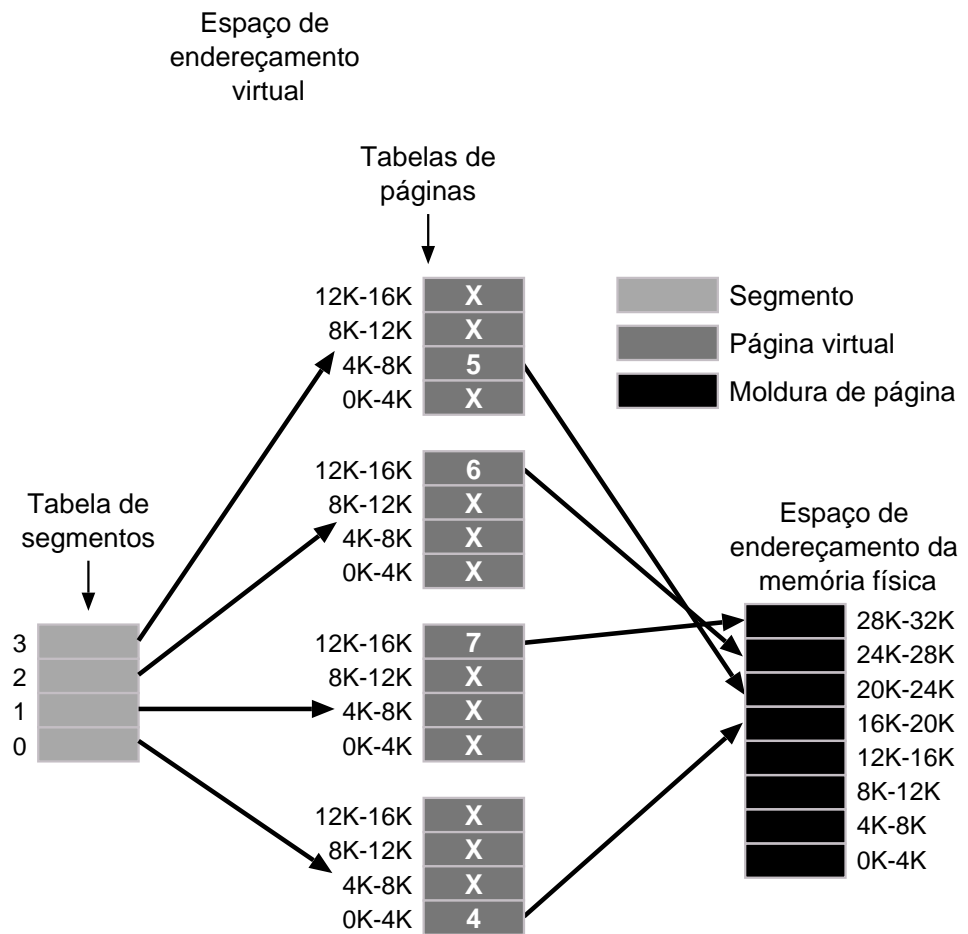
Baseado nos resultados anteriores, a resposta à questão é sim, é verdade que a mesma página virtual, a 1, é a escolhida por todos os algoritmos para ser substituída caso ocorra uma nova falha de página.

4. (3,0) Suponha que um computador utilize a técnica de segmentação com paginação. Na figura a seguir mostramos as tabelas de páginas para os 4 segmentos de 16KB do computador. Inicialmente nenhuma página de nenhum segmento está na memória, ou seja, todas as molduras de página estão vazias. Como a tabela da figura será preenchida, se:



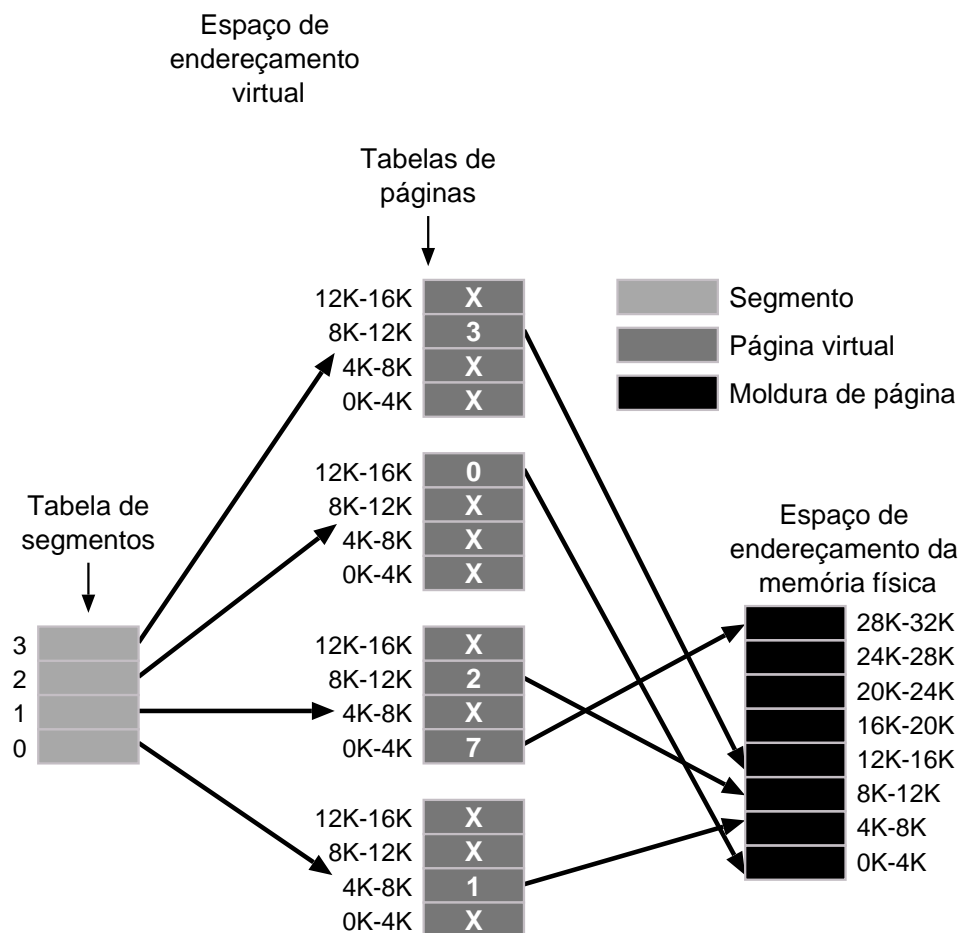
- (a) (1,5) Os endereços (1, 13122), (2, 16300), (3, 7593), (0, 2012) e (2, 12289) forem acessados e as molduras forem alocadas em ordem decrescente de acordo com os seus números?

Resp.: Como podemos ver pela figura, os endereços virtuais (1, 13122), (2, 16300), (3, 7593), (0, 2012) e (2, 12289) estão, respectivamente, nas páginas virtuais 3 do segmento 1, 3 do segmento 2, 1 do segmento 3, 0 do segmento 0, e 3 do segmento 2. Como as molduras de página são alocadas em ordem decrescente de acordo com os seus números, teremos o mapeamento dado na figura a seguir:



- (b) (1,5) Os endereços (0, 7890), (1, 0), (2, 16383), (3, 12287) e (1, 8888) forem acessados e as molduras forem alocadas na ordem 1, 7, 0, 3, 2, 5, 6 e 4?

Resp.: Como podemos ver pela figura, os endereços virtuais (0, 7890), (1, 0), (2, 16383), (3, 12287) e (1, 8888) estão, respectivamente, nas páginas virtuais 1 do segmento 0, 0 do segmento 1, 3 do segmento 2, 2 do segmento 3, e 2 do segmento 1. Ao mapearmos as molduras de acordo com a ordem dada no enunciado, teremos o mapeamento dado na figura a seguir:



5. (2,0) Suponha que um disco tenha e endereços e que o sistema operacional use a técnica de alocação por lista encadeada utilizando um índice. Responda, justificando a sua resposta.

(a) (1,0) Quantos bits são gastos pela tabela usada nessa técnica?

Resp.: Como o disco tem e endereços, então gastaremos $\lceil \log_2 e \rceil$ bits para representar um endereço. Agora, como existe uma entrada na tabela para cada endereço, o número de bits é igual a $e \lceil \log_2 e \rceil$.

- (b) (1,0) Suponha que o espaço necessário para armazenar a tabela do item (a) ocupe 6,25% do tamanho de um bloco do disco. Qual é o tamanho total do disco em bytes?

Resp.: Como a tabela, com $e \lceil \log_2 e \rceil$ bits, ocupa 6,25% ou $1/16$ do tamanho do bloco, então esse bloco tem tamanho igual a $16e \lceil \log_2 e \rceil$ bits ou $2e \lceil \log_2 e \rceil$ bytes. Agora, como temos e endereços no disco, ou seja, e blocos, pois cada endereço está associado a um bloco diferente, então o tamanho do disco é igual a $2e^2 \lceil \log_2 e \rceil$ bytes.