



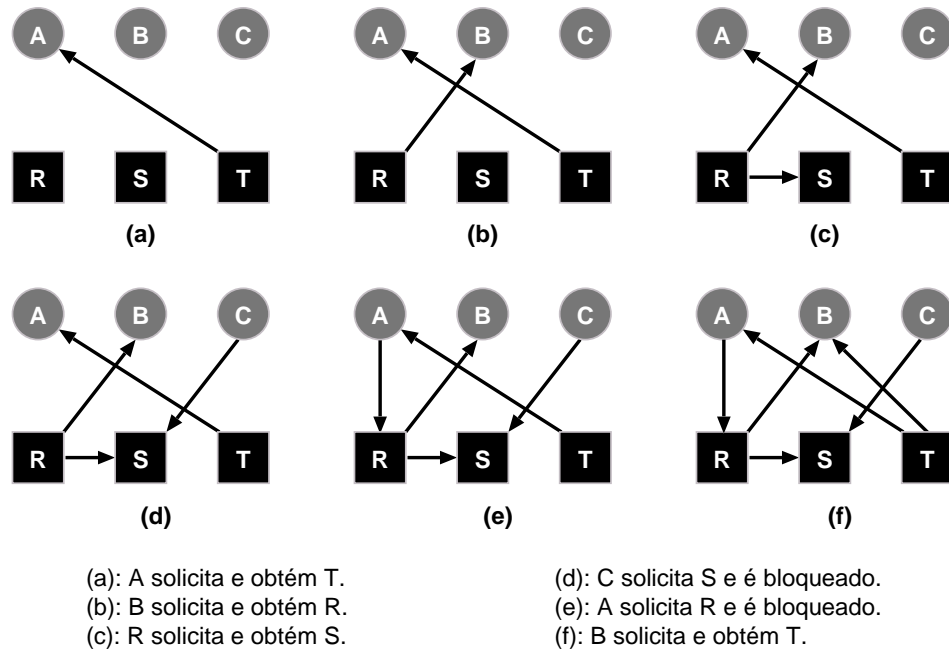
Curso de Tecnologia em Sistemas de Computação
Disciplina de Sistemas Operacionais
Professores: Valmir C. Barbosa e Felipe M. G. França
Assistente: Alexandre H. L. Porto

Quarto Período
Gabarito da AD2 - Primeiro Semestre de 2015

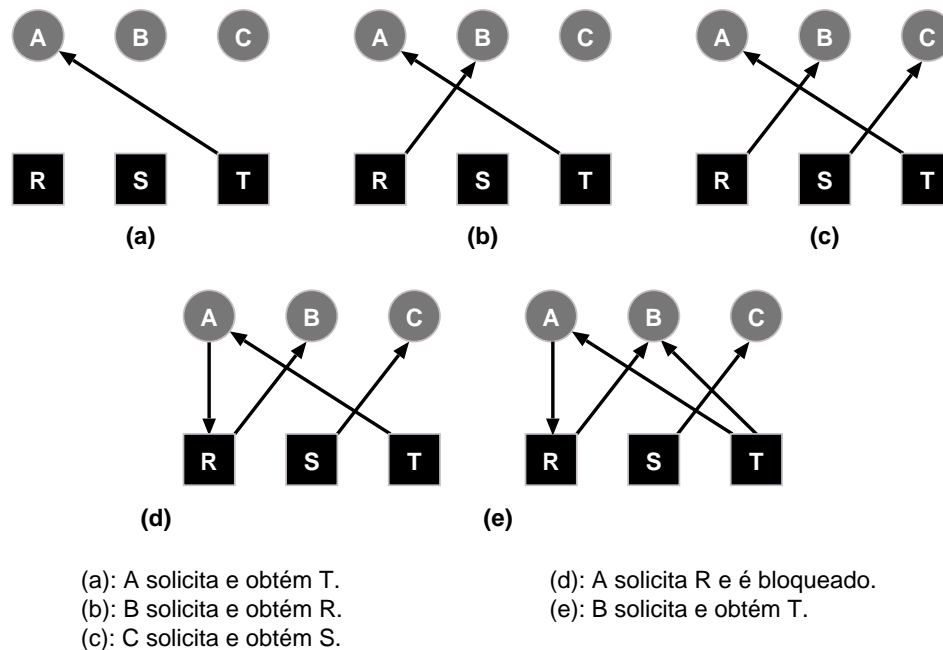
Atenção: Cada aluno é responsável por redigir suas próprias respostas. Provas iguais umas às outras terão suas notas diminuídas. As diminuições nas notas ocorrerão em proporção à similaridade entre as respostas. Exemplo: Três alunos que respondam identicamente a uma mesma questão terão, cada um, $1/3$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

Nome -
Assinatura -

-
1. (1,0) Um aluno de Sistemas Operacionais disse que a figura a seguir é um possível exemplo de uma sequência de alocações de recursos, sendo que R e S são recursos não-preemptivos, T é um recurso preemptivo, e A , B e C são processos. A sequência do aluno está correta? Se você achar que sim, basta dizer isso mas, se você achar que não, diga quais foram os passos errados do aluno na sequência. Justifique a sua resposta.



Resp.: A sequência do aluno não está correta. O passo (c) não tem sentido, porque como R e S são recursos, então R não pode ter obtido S com sucesso devido a somente processos poderem obter recursos. Além disso, devido a esse erro, o processo C não deveria ter sido bloqueado no passo (d) ao obter S porque, como não é possível R obter S, então S estaria livre e C poderia obtê-lo sem ser bloqueado. Na figura a seguir apresentamos uma das possíveis sequências corretas.



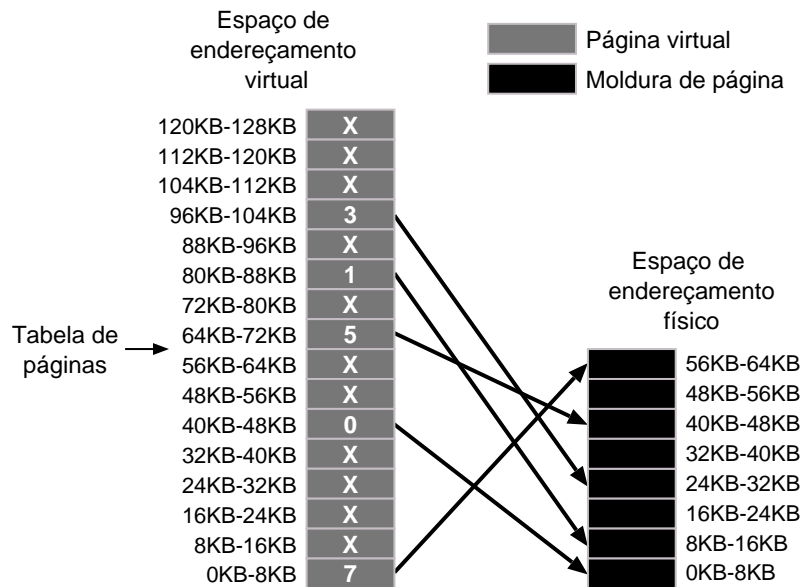
2. (3,0) Suponha que um computador tenha uma memória virtual de 128KB e uma memória física de 64KB. Suponha ainda que as páginas virtuais tenham 8KB de tamanho e que, inicialmente, nenhuma página virtual esteja mapeada em uma moldura de página. Responda:

(a) (1,0) Se o processo acessar os endereços virtuais dados na parte esquerda da tabela a seguir, e se cada endereço virtual dado em uma linha da tabela for mapeado para o endereço físico dado na coluna ao lado, como será o mapeamento das páginas virtuais nas molduras de página, usando uma figura similar à dada na transparência 19 da aula 8?

Endereço virtual	Endereço físico
100000	26272
65536	40960
87654	13926
49151	8191
0	57344

Resp.: Pelo enunciado da questão, vemos que o espaço de endereçamento virtual possui 128KB, com endereços de 17 bits variando de 0 até 131071. Já o espaço de endereçamento físico possui 64KB, com endereços de 16 bits variando de 0 até 65535. As páginas possuem 8KB de tamanho e, com isso, temos 16 páginas virtuais e 8 molduras de página. Logo, um endereço virtual é dividido em 4 bits para o número da página virtual e 13 bits para o deslocamento, e um endereço físico é dividido em 3 bits para o número da moldura de página e 13 bits para o deslocamento. A seguir apresentamos uma nova versão da tabela mostrando, separadas por uma barra vertical, a divisão de cada endereço virtual nos campos número da página virtual e deslocamento, e a divisão de cada endereço físico nos campos número da moldura de página e deslocamento (note que todos os deslocamentos são coerentes, o que indica que o mapeamento da tabela está correto). Usando os campos com os números das páginas virtuais e os campos com os números das molduras de página obtemos a figura dada a seguir, após a tabela e similar à dada na aula, com os mapeamentos.

Endereço virtual	Endereço físico
100000 (1100 0011010100000)	26272 (011 0011010100000)
65536 (1000 0000000000000)	40960 (101 0000000000000)
87654 (1010 1011001100110)	13926 (001 1011001100110)
49151 (0101 1111111111111)	8191 (000 1111111111111)
0 (0000 0000000000000)	57344 (111 0000000000000)



- (b) (1,0) Considerando o mapeamento obtido no item (a), existe algum endereço virtual mapeado no endereço físico 123456?

Resp.: Não, porque o endereço físico 123456 não é válido, devido a faixa de endereços, em uma memória física com 64KB, ser de 0 até 65535.

- (c) (1,0) Considerando novamente o mapeamento obtido no item (a), o que ocorreria se o endereço virtual a fosse acessado, sendo $77824 \leq a \leq 86016$?

Resp.: A faixa de possíveis endereços virtuais faz com que a possa pertencer a página virtual 9 ou a página virtual 10. Se a estiver na página virtual 10, que começa no endereço virtual 81920, como a página 10 está na memória e mapeada, segundo o mapeamento obtido no item (a), na moldura de página 1, que começa no endereço físico 8192 então, se $81920 \leq a \leq 86016$, o endereço físico correspondente a a será $8192 + a - 81920$, ou seja, $a - 73728$. Porém, se a pertencer a página virtual 9, que começa no endereço virtual 73728, ou seja, se $77824 \leq a \leq 81919$, uma falha de página será gerada quando o endereço virtual for aces-

sado, porque a página 9 não está mapeada em nenhuma moldura de página. Na tabela a seguir mostramos o endereço físico obtido para cada uma das possíveis molduras de página não usadas, lembrando que a moldura de página m começa no endereço físico $8192m$ e que, para essa moldura, o endereço físico correspondente a a será $8192m + a - 73728$:

Número da moldura de página	Endereço físico
2	$a - 57344$
4	$a - 40960$
6	$a - 24576$

3. (1,5) Suponha que um processo tenha acessado, em ordem, as páginas virtuais 1, 5, 2, 1, 4, 3, 2, 1, 6 e 3, e que tenham sido alocadas 3 molduras, inicialmente vazias, para o processo. Se o bit **referenciada** de uma página for ligado quando ela for copiada para a memória e permanecer ligado até três páginas terem sido acessadas depois dela, quantos acessos a páginas não gerarão falhas de página se o algoritmo de segunda chance for usado? E se o algoritmo LRU passar a ser usado, supondo que agora tenham sido alocadas 4 molduras, inicialmente vazias, ao processo? Justifique a sua resposta.

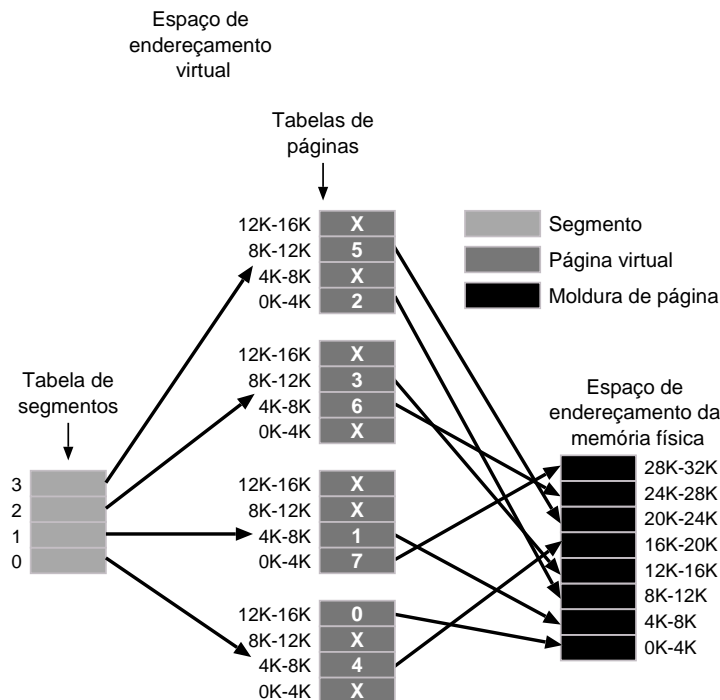
Resp.: -Primeiramente vamos mostrar como as páginas são substituídas de acordo com o algoritmo de segunda chance, usando o critério de atualização do bit referenciada dado no enunciado da questão. Na tabela dada a seguir mostramos, em cada linha, o que ocorre ao acessarmos as páginas na ordem dada no enunciado. Para cada uma dessas linhas mostramos na primeira coluna a página que é acessada. Já na segunda coluna mostramos a ordem em que as páginas devem ser escolhidas, sendo que ao lado da página, em parênteses, mostramos o valor atual do bit referenciada. Finalmente, na última coluna, mostramos se a página gereou uma falha de página ao ser acessada. Pela tabela, vemos que 3 acessos não geraram falhas de página.

Página	Ordenação	Falha?
1	1(1)	Sim
5	1(1) 5(1)	Sim
2	1(1) 5(1) 2(1)	Sim
1	1(0) 5(1) 2(1)	Não
4	5(0) 2(1) 4(1)	Sim
3	2(0) 4(1) 3(1)	Sim
2	2(0) 4(1) 3(1)	Não
1	4(0) 3(1) 1(1)	Sim
6	3(0) 1(1) 6(1)	Sim
3	3(0) 1(1) 6(1)	Não

-Agora, se o algoritmo LRU for usado com quatro molduras de página, a tabela a seguir mostra o que ocorrerá ao acessarmos as páginas na ordem dada no enunciado. A tabela é similar a dada anteriormente para o algoritmo de segunda chance mas, agora, a segunda coluna mostra somente a ordem segundo a qual as páginas devem ser escolhidas, já que o algoritmo LRU não usa o valor do bit referenciada. Pela tabela, vemos que agora 4 acessos não geraram falhas de página.

Página	Ordenação	Falha?
1	1	Sim
5	1 5	Sim
2	1 5 2	Sim
1	5 2 1	Não
4	5 2 1 4	Sim
3	2 1 4 3	Sim
2	1 4 3 2	Não
1	4 3 2 1	Não
6	3 2 1 6	Sim
3	2 1 6 3	Não

4. (1,5) Suponha que o sistema operacional use a técnica de segmentação com paginação, e que o computador tenha um espaço de endereçamento virtual dividido como na figura dada a seguir. Responda, justificando a sua resposta:



- (a) (0,5) Para cada segmento, que faixas de endereços geram falhas de página ao serem acessadas, sendo que “X” em uma entrada de uma tabela de páginas indica que a página não está na memória?

Resp.: Para o segmento 0, vemos que uma falha de página ocorrerá ao acessarmos os endereços de 0 até 4095 (faixa 0K-4K) ou os endereços de 8192 até 12287 (faixa 8K-12K). Para o segmento 1, uma falha de página será gerada se acessarmos os endereços de 8192 até 16383 (faixas 8K-12K e 12K-16K). Para o segmento 2, uma falha de página será gerada ao acessarmos os endereços de 0 até 4095 (faixa 0K-4K) ou os endereços de 12288 até 16383 (faixa 12K-16K). Finalmente, para o segmento 3, os acessos aos endereços de 4096 até 8191 (faixa 4K-8K) ou aos endereços de 12288 até 16383 (faixa 12K-16K) gerarão falhas de página.

- (b) (1,0) Suponha que um processo A esteja usando o segmento 1, que um processo B esteja usando os segmentos 2 e 3, e que um processo C esteja usando o segmento 0. Suponha ainda que a ordem

de carga das páginas nas molduras tenha sido a seguinte: 5, 2, 1, 4, 3, 6, 7 e 0. Qual será a página substituída quando o processo C acessar um endereço que não esteja na memória, se o sistema operacional usar o algoritmo FIFO para substituir as páginas com uma política de alocação global? E se a política for local ao invés de global e agora o processo A tentar acessar um endereço que não esteja na memória?

Resp.: Como vimos na aula 9, no algoritmo FIFO, a página copiada há mais tempo será escolhida para ser substituída. Como podemos ver pela figura, as molduras 1 e 7 estão alocadas ao segmento 1 usado pelo processo A, as molduras 2, 3, 5 e 6 estão alocadas aos segmentos 2 e 3 usados pelo processo B, e as molduras restantes, 0 e 4, estão alocadas ao segmento 0 usado pelo processo C. Temos, então:

- Se a política de alocação global for usada e C gerar uma falha de página, todas as molduras deverão ser consideradas pelo algoritmo FIFO; Agora, como a ordem 5, 2, 1, 4, 3, 6, 7 e 0 está também de acordo com a ordem crescente do tempo de cópia das páginas, então a página 2 do segmento 3 (associado ao processo B), que está mapeada na moldura 5, será a escolhida para ser substituída;
- Se a política de alocação for a local e agora A gerar uma falha de página, então somente as molduras associadas ao processo A, ou seja, as molduras 1 e 7, deverão ser consideradas; Como neste caso a ordenação de acordo com o tempo da cópia é 1 e 7, então a página 1 do segmento 1, que está mapeada na moldura 1, será a escolhida para ser substituída.

5. (1,5) Suponha que dois arquivos, A e B, estejam armazenados em um disco com n blocos, que A esteja armazenado, em ordem, nos blocos x_i , $1 \leq i \leq a$, e que B esteja armazenado, em ordem, nos blocos y_j , $1 \leq j \leq b$. Responda as seguintes perguntas, justificando a sua resposta:

- (a) (1,0) Se a alocação contígua for usada, a que restrições os blocos de A e B deverão satisfazer?

Resp.: Se $x_1 < y_1$, ou seja, se A começar antes de B no disco, então A deve terminar em um bloco antes do início de B, ou seja, $x_a < y_1$. Além disso, devemos garantir que A e B usem somente blocos válidos, numerados de 0 até $n - 1$, porque o disco tem n blocos, o que implica que $x_1 \geq 0$ e $y_b < n$. Logo, nesse caso, $0 \leq x_1 < x_a < y_1 < y_b < n$. No caso contrário (ou seja, se $x_1 > y_1$, porque $x_1 = y_1$ implicaria em uma inconsistência devido a A e B começarem no mesmo bloco), então B deve terminar em um bloco antes do início de A, ou seja, $y_b < x_1$. Além disso, para garantir que A e B usem blocos válidos, $y_1 \geq 0$ e $x_a < n$. Portanto, nesse último caso, $0 \leq y_1 < y_b < x_1 < x_a < n$.

- (b) (0,5) Se a alocação por lista encadeada utilizando um índice for usada, qual será o tamanho em bits da tabela usada pela alocação?

Resp.: Como vimos na aula 11, a tabela da alocação por lista encadeada utilizando um índice possui uma entrada para cada bloco e, como cada entrada precisa armazenar o número de um bloco, o número de bits de cada entrada é igual ao número de bits necessário para representar o número de um bloco. No caso de um disco com n blocos numerados de 0 até $n - 1$, vamos precisar de $\lceil \log_2 n \rceil$ bits para representar cada entrada da tabela. Agora, como temos n entradas, então o número total de bits da tabela é de $n \lceil \log_2 n \rceil$ bits.

6. (1,5) Suponha que três arquivos, A, B e C, estejam armazenados no disco do computador. O arquivo A está armazenado a partir do bloco que está a $2/5$ do início do disco e está usando 40% do disco. O arquivo B está armazenado a partir do bloco central do disco e está usando 10% do disco. Finalmente, o arquivo C está armazenado a partir do bloco que está a $2/5$ do final do disco e está usando 20% do disco. Suponha ainda que 30% dos blocos do disco estejam marcados como livres. Dado que está sendo utilizada a alocação contígua, existe alguma in-

consistência no sistema de arquivos? Justifique a sua resposta.

Resp.: A seguir mostramos uma figura com a disposição dos arquivos A, B e C no disco, sendo que os blocos do disco são numerados da esquerda para a direita. Como o arquivo A está a $2/5$, ou seja, 40%, do início do disco e usa 40% do disco, então o primeiro bloco após A está a 20% do final do disco. Logo, todos os blocos de B são compartilhados com A porque B começa no meio ou a 50% do início (ou do final) do disco e porque, como B usa 10% do disco, o primeiro bloco após B está a 60% do início ou a 40% do final do disco. Além disso, todos os blocos de C também são compartilhados com o A, porque como C começa a $2/5$ do final do disco ou a $3/5$ do início do disco, ou seja, a 60% do início do disco, e como C usa 20% do disco, C termina no mesmo bloco que A no disco. Note que B e C não compartilham blocos, porque C começa no bloco logo após o último bloco de B no disco. Finalmente, como somente 30% dos blocos do disco são marcados como livres e, como pela figura vemos que 60% dos blocos do disco estão livres, então 30% desses blocos estão ausentes. Logo, temos dois tipos de inconsistência no sistema de arquivos: (i) existem blocos alocados a mais de um arquivo e (ii) existem blocos ausentes, não marcados como livres e nem usados pelos arquivos A, B ou C.

