



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

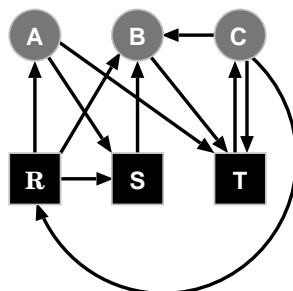
Quarto Período
AD2 - Primeiro Semestre de 2010

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

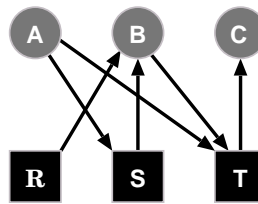
Atenção: Tem havido muita discussão sobre a importância de que cada aluno redija suas próprias respostas às questões da AD2. Os professores da disciplina, após refletirem sobre o assunto, decidiram o seguinte: Cada aluno é responsável por redigir suas próprias respostas. Provas iguais umas às outras 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. (2.0) Modifique o grafo dado a seguir de modo a que ele seja um grafo de recursos correto e sem impasses.



Resp.: Esta questão tem diversas respostas, mas você não pode escolher qualquer grafo com os processos A, B e C e os recursos R, S e T. O seu grafo deve ser um grafo de recursos correto (segundo os critérios vistos na Aula 7), deve ser livre de impasses, e deve ser uma modificação do grafo dado na questão. O grafo dado na questão tem diversos erros, que precisam ser removidos para obtermos um grafo de recursos correto. Os dois primeiros erros são que não podem existir arestas entre vértices que representem processos (a aresta $C \rightarrow B$), ou entre vértices que representem recursos (a aresta $R \rightarrow S$). O terceiro erro é que existem duas arestas se originando do recurso R, significando que R pertence ao mesmo tempo aos processos A (a aresta $R \rightarrow A$) e B (a aresta $R \rightarrow B$), o que não é possível, porque um recurso não pode pertencer simultaneamente a mais de um processo. Finalmente, o último erro é que o grafo tem as arestas $C \rightarrow T$ e $T \rightarrow C$, o que significa que C está tentando obter um recurso que já possui. Além disso, como não sabemos se os recursos R, S e T são ou não preemptivos, então o grafo não pode ter ciclos orientados, porque somente a ausência destes ciclos garantirá a não ocorrência de um impasse. O grafo dado a seguir é uma das possíveis modificações do grafo original. Neste grafo, definimos que o recurso R pertence ao processo B (com isso, removemos a aresta $R \rightarrow A$), e que o processo C possui o recurso T (devido a isso, também removemos a aresta $C \rightarrow T$). Também removemos as arestas $C \rightarrow B$ e $R \rightarrow S$ que não podem existir em um grafo de recursos. Finalmente, escolhemos remover a aresta $C \rightarrow R$, para evitar a formação do ciclo orientado $R \rightarrow B \rightarrow T \rightarrow C \rightarrow R$.



2. (2.0) Dada a tabela de páginas a seguir, diga quais são os possíveis endereços físicos correspondentes ao endereço virtual 43338.

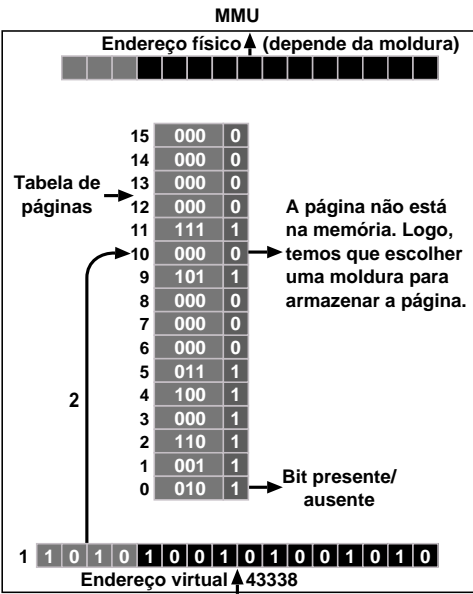
MMU

Endereço físico correspondente	
15	000 0
14	000 0
13	000 0
12	000 0
11	111 1
10	000 0
9	101 1
8	000 0
7	000 0
6	000 0
5	011 1
4	100 1
3	000 1
2	110 1
1	001 1
0	010 1

Endereço virtual 43338

Resp.: Na figura a seguir mostramos o que ocorre quando o sistema operacional tenta converter o endereço virtual 43338. Como podemos ver pela figura, este endereço pertence à página virtual 10, que não está armazenada na memória, porque o seu bit presente/ausente é 0. Neste caso, antes de o processador finalizar a conversão do endereço virtual para o endereço físico (e, com isso, executar os passos 3 e 4 vistos na aula 8), o sistema operacional precisará escolher qual moldura de página deverá armazenar esta página virtual. Como não sabemos o critério usado pelo sistema para escolher as molduras de página, então precisaremos dar todas as possibilidades, porque qualquer moldura pode ser a escolhida para armazenar a página. Logo, os possíveis endereços virtuais serão os dados na tabela dada a seguir, sendo que cada endereço obtido

nesta tabela está associado a uma das possíveis molduras de página. Em cada coluna desta tabela damos, entre parênteses, o valor binário correspondente ao valor decimal.



Página virtual	Endereço físico
0 (000)	2378 (000100101001010)
1 (001)	6474 (001100101001010)
2 (010)	10570 (010100101001010)
3 (011)	14666 (011100101001010)
4 (100)	18762 (100100101001010)
5 (101)	22858 (101100101001010)
6 (110)	26954 (110100101001010)
7 (111)	31050 (111100101001010)

3. (2.0) Suponha que um processo tenha acessado as páginas virtuais na ordem 2, 1, 2, 4, 2, 4, e suponha que duas molduras, inicialmente vazias, tenham sido alocadas a ele. Um aluno de sistemas operacionais afirmou que se o processo depois acessasse a página 5, a página 2 seria a substituída independentemente de o sistema operacional usar o algoritmo LRU ou a sua implementação em hardware baseada em uma

matriz. A afirmação do aluno está correta? Justifique a sua resposta.

Resp.: Ao usarmos o algoritmo LRU, obteremos a tabela dada a seguir, sendo que cada uma das linhas está associada a uma das páginas acessadas, na ordem dada no enunciado. Na primeira coluna da tabela damos a página acessada. Na segunda coluna mostramos a ordem em que as páginas são escolhidas, de acordo com o tempo em que elas foram acessadas pela última vez. Finalmente, na terceira coluna, dizemos se o acesso gerou ou não uma falha de página. Pela última linha da tabela, vemos que a página 2 será a próxima a ser substituída se uma nova falha de página ocorrer.

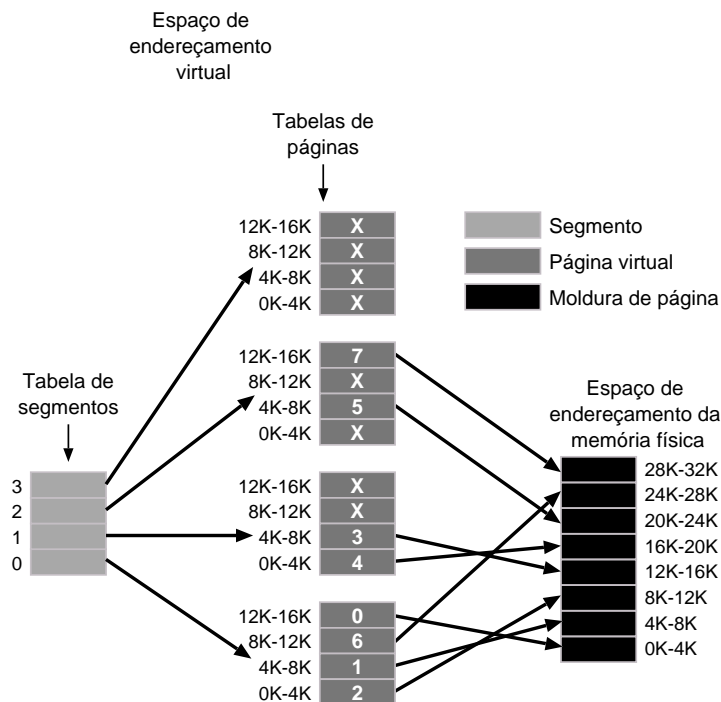
Página	Ordem de escolha		Ocorreu uma falha?
2	2		Sim
1	2	1	Sim
2	1	2	Não
4	2	4	Sim
2	4	2	Não
4	2	4	Não

Quando a implementação do LRU em hardware for usada pelo sistema operacional, uma matriz quadrada 2×2 será usada, porque 2 molduras foram disponibilizadas ao processo. Na figura a seguir mostramos a sequência de passos executados ao usarmos este algoritmo. Na parte (a) da figura mostramos o estado da matriz antes de acessarmos qualquer página e, nas partes de (b) até (g) desta mesma figura, a matriz obtida após acessarmos cada uma das páginas, na ordem dada no enunciado da questão. Em cada parte da figura, com exceção das partes (a) e (b), destacamos, em negrito, a página a ser substituída e a linha da matriz associada à página a ser substituída. Nesta figura, também supomos que a página virtual 2 foi copiada na moldura de página 0 quando a acessamos pela primeira vez. Como podemos ver pela parte (g) da figura, a página 2 também será a substituída caso ocorra uma nova falha de página. Agora, como o algoritmo LRU também escolheu a página 2, então a afirmação do aluno está correta. Note que o resultado seria

o mesmo se tivéssemos colocado a página 2 inicialmente na moldura 1, ou seja, obteríamos o mesmo padrão em que duas matrizes se alternam, com a única diferença de que o padrão começaria com a matriz $\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$ ao invés da matriz $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$, o que significaria que a página 2 ainda seria a substituída.

Matriz	Matriz	Matriz	Matriz
0 1	0 1	0 1	0 1
0 0 0	0 0 1	0 0 0	0 0 1
1 0 0	1 0 0	1 1 0	1 0 0
Páginas	Páginas	Páginas	Páginas
0 1	0 1	0 1	0 1
- -	2 -	2 1	2 1
Falha: -	Falha: Sim	Falha: Sim	Falha: Não
(a)	(b)	(c)	(d)
Matriz	Matriz	Matriz	
0 1	0 1	0 1	
0 0 0	0 0 1	0 0 0	
1 1 0	1 0 0	1 1 0	
Páginas	Páginas	Páginas	
0 1	0 1	0 1	
2 4	2 4	2 4	
Falha: Sim	Falha: Não	Falha: Não	
(e)	(f)	(g)	

- (2.0) 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. Usando essas tabelas, forneça os endereços virtuais bidimensionais (segmento, endereço dentro do segmento) correspondentes aos seguintes endereços físicos:



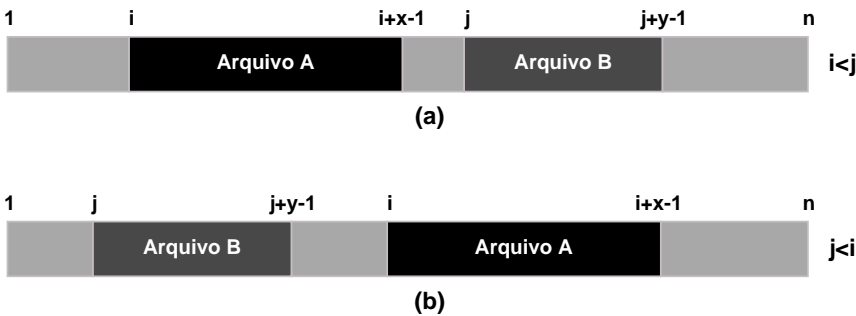
- (a) (0.7) 12321.
 (b) (0.7) 32000.
 (c) (0.6) 1974.

Resp.: Pela figura vemos que o espaço de endereçamento de cada segmento possui 16K, com endereços de 14 bits variando de 0 até 16383. Já o espaço de endereçamento físico possui 32K, com endereços variando de 0 até 32767. Como as páginas possuem 4K de tamanho, cada segmento é dividido em 4 páginas virtuais, e o espaço de endereçamento físico é dividido em 8 molduras de página. Logo, um endereço físico é dividido do seguinte modo: 3 bits para o número da moldura de página e 12 bits para o deslocamento. A seguir mostramos como converter cada um dos endereços físicos para os endereços virtuais bidimensionais correspondentes. Mostramos também, entre parênteses, cada endereço físico na base binária, separando os dois campos do endereço pelo caractere “|”.

- (a) Endereço físico 12321 (011 | 000000100001): para este endereço, vemos que estamos acessando a palavra 33 da moldura de página 3 (com endereços de 12288 até 16383). Agora, como a página virtual 1 do segmento 1 está mapeada nesta moldura (com endereços de 4096 até 8191), então a segunda componente do endereço virtual bidimensional é 4096 (a primeira palavra da página) mais 33 (o deslocamento), isto é, 4129. Logo, o endereço virtual bidimensional é (1, 4129).
 - (b) Endereço físico 32000 (111 | 110100000000): neste caso, vemos que estamos acessando a palavra 3328 da moldura de página 7 (com endereços de 28672 até 32767). Pela figura, vemos que a página virtual 3 do segmento 2 está mapeada nesta moldura (com endereços de 12288 até 16383) e, com isso, a segunda componente do endereço virtual bidimensional é 12288 (a primeira palavra da página) mais 3328 (o deslocamento), isto é, 15616. Logo, o endereço virtual bidimensional é (2, 15616).
 - (c) Endereço físico 1974 (000 | 011110110110): este endereço aponta para a palavra 1974 da moldura de página 0 (com endereços de 0 até 4095). Agora, como a página virtual 3 do segmento 0 está mapeada nesta moldura (com endereços de 12288 até 16383), então a segunda componente do endereço virtual bidimensional é 12288 (a primeira palavra da página) mais 1974 (o deslocamento), isto é, 14262. Logo, o endereço virtual bidimensional é (0, 14262).
5. (2.0) Suponha que o disco do computador tenha n blocos e que o sistema operacional use a alocação contígua ao alocar os blocos aos arquivos. Suponha ainda que dois arquivos, A e B, tenham sido alocados neste disco, ocupando, respectivamente, x e y blocos, sendo que $x + y \leq n$. Qual será o tamanho máximo que um novo arquivo C poderá ter em função das posições iniciais de A e B? Justifique a sua resposta.

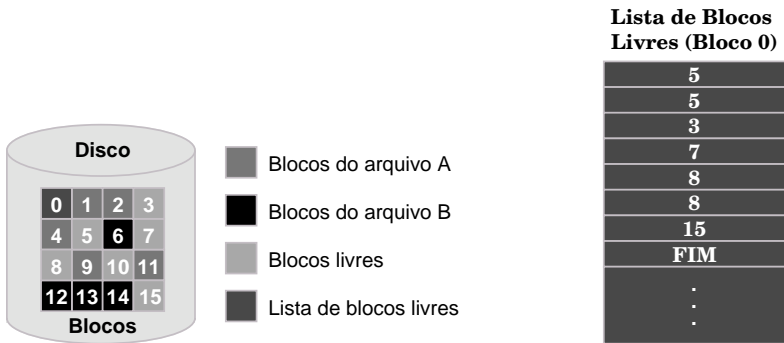
Resp.: Suponha que i é a posição inicial do arquivo A no disco e que j é a posição inicial do arquivo B no mesmo disco. Nesta questão, vamos mostrar a seguir duas possibilidades, $i < j$ e $j < i$ (note que $i \neq j$, pois o sistema operacional não pode armazenar os dois arquivos a partir da mesma posição) mas, como podemos supor, sem perda de generalidade, que $i < j$ (ou que $j < i$), então só é necessário ao

responder a questão dar uma das duas possibilidades. Na parte (a) da figura a seguir mostramos o que ocorre quando $i < j$. Neste caso, o número máximo de blocos que o novo arquivo C poderá ter será o maior valor dentre $i - 1$, $j - i - x + 1$ e $n - j - y + 1$, ou seja, $\max\{i - 1, j - i - x + 1, n - j - y + 1\}$. Já na parte (b) da figura mostramos o que ocorre quando $j < i$. Neste último caso, o número máximo de blocos do arquivo C será o maior valor dentre $j - 1$, $i - j - y + 1$ e $n - i - x + 1$, ou seja, $\max\{j - 1, i - j - y + 1, n - i - x + 1\}$.



(b)

6. (2.0) Considere um computador com um disco de 16 blocos e a alocação representada pela figura a seguir. Supondo que o bloco 0 seja reservado e armazene a lista de blocos livres, e que os arquivos A e B sejam os únicos arquivos armazenados no disco, responda às seguintes perguntas:



Obs.: Esta questão 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.

- (a) (1.0) O sistema de arquivos está consistente? Justifique a sua resposta.

Resp.: Não, pois temos dois tipos de inconsistências no sistema de arquivos dado na figura. A primeira delas é que os blocos 5 e 8 estão duplicados na lista de blocos livres. A segunda é que o bloco 10 é um bloco ausente, porque não está na lista de blocos livres e nem está alocado aos arquivos A e B. Note que o bloco 0 não pode estar nesta lista, pois ele está reservado para armazená-la.

- (b) (1.0) Suponha que os blocos sejam sempre escolhidos na ordem definida, de cima para baixo, na lista de blocos livres dada na figura. O que ocorrerá se o arquivo A aumentar o seu tamanho em um bloco, e depois o arquivo B também aumentar o seu tamanho em um bloco? Justifique a sua resposta.

Resp.: Como o bloco 5 está duplicado na lista de blocos livres, e como ele está ocupando as duas primeiras posições desta lista, então ele será alocado, ao mesmo tempo, aos arquivos A e B. Com isso, o conteúdo do arquivo A será comprometido após o arquivo B salvar os seus dados no bloco 5, porque os dados salvos pelo arquivo A neste bloco serão perdidos. Além disso, se futuramente alterarmos um dos arquivos e salvarmos novos dados neste bloco, o conteúdo do outro arquivo também será comprometido. Logo, pelos problemas descritos anteriormente, o que ocorrerá é que não será mais confiável armazenar dados nos arquivos A e B.