



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 AP2 - Segundo Semestre de 2019

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. (1,5) Suponha que n recursos não-preemptivos possam ser usados por três processos A, B e C. Quantos impasses diferentes existirão envolvendo dois desses processos? Justifique a sua resposta.

Resp.: Qualquer impasse envolvendo dois processos deverá envolver também dois recursos não-preemptivos. Como temos três processos, A, B e C, então três pares de processos poderão fazer parte de um impasse: A e B, A e C ou B e C. Agora, para cada par, um dos processos poderá solicitar com sucesso um dos n recursos e o outro processo um dos $n - 1$ recursos ainda não obtidos. Depois disso, cada processo do par deverá tentar obter o recurso que o outro processo do par possui para poder ser bloqueado. Logo, para cada par de processos, existirão $n(n - 1)$ impasses diferentes. Finalmente, como temos três pares, então o número total de impasses diferentes com dois processos e dois recursos será $3n(n - 1)$.

2. (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 existir DMA no hardware, não será mais necessário desperdiçar tempo de processamento com as transferências de dados entre a controladora de um dispositivo e a memória, porque a controladora poderá acessar diretamente a memória para transferir os dados.

Resp.: V (Verdadeira).

- (b) (0,5) O gerenciamento de memória por *overlays* facilita o desenvolvimento de programas com tamanho maior do que o tamanho da memória física, pois é o sistema operacional, e não o programador, o responsável por dividir o programa em *overlays* diferentes.

Resp.: F (Falsa), pois o maior problema do gerenciamento por *overlays* é exatamente a complexidade de desenvolver um programa, já que o programador é o responsável por dividir o programa em *overlays* diferentes.

- (c) (0,5) O algoritmo de idade é uma possível implementação do algoritmo NRU, pois é complicado manter as quatro classes, usadas pelo algoritmo NRU, definidas de acordo com os bits **modificada** e **referenciada**.

Resp.: F (Falsa), pois como o algoritmo de idade é uma implementação do algoritmo LRU então, além de não serem usadas as classes, a complexidade do algoritmo LRU vem do fato de ser necessário manter uma lista encadeada com as páginas ordenadas de acordo com o tempo do seu último acesso.

- (d) (0,5) A principal diferença entre as políticas de alocação igualitária e proporcional é que, na primeira, as molduras são igualmente divididas entre os processos e, na última, as molduras são divididas de acordo com o tamanho dos processos.

Resp.: V (Verdadeira).

- (e) (0,5) A principal vantagem da alocação por lista encadeada em relação à contígua é que, na primeira, um arquivo poderá usar menos blocos do disco para ser armazenado.

Resp.: F (Falsa), pois é exatamente o oposto, ou seja, mais blocos poderão ser usados para armazenar o arquivo, porque parte do espaço de cada bloco será usada para armazenar o número do próximo bloco do arquivo no disco.

3. (1,5) Suponha que um processo tenha acessado, em ordem, as páginas 2, 1, 2, 3, 2, 1, 3 e 1. Suponha ainda que o algoritmo LRU seja usado pelo sistema operacional, e que duas molduras, inicialmente vazias, estejam disponíveis para o processo. Quantas falhas de página serão geradas pelo processo após acessar todas as páginas? Se agora o algoritmo FIFO passar a ser usado pelo sistema operacional, o número de falhas será o mesmo? Justifique a sua resposta.

Resp.: -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 a seguir mostramos o que ocorrerá quando as páginas usadas pelo processo forem acessadas na ordem dada no enunciado. Para cada linha da tabela mostramos na primeira coluna a página que é acessada, na segunda coluna a ordem em que as páginas devem ser escolhidas e, na última coluna, se o acesso à página gerou ou não uma falha de página. Como podemos ver pela tabela, ocorrerão 5 falhas de página após o processo acessar todas as páginas.

Página	Ordenação		Falha?
2	2		Sim
1	2	1	Sim
2	1	2	Não
3	2	3	Sim
2	3	2	Não
1	2	1	Sim
3	1	3	Sim
1	3	1	Não

-Agora, como vimos na aula 9, no algoritmo FIFO, as páginas são primeiramente ordenadas, em ordem crescente, de acordo com o tempo da sua cópia para a memória. A página a ser substituída é a primeira página segundo essa ordenação, isto é, a página copiada há mais tempo para a memória. A tabela a seguir é similar à anterior, mas com a diferença de que a segunda coluna agora mostra a ordem segundo o algoritmo FIFO. Como podemos ver pela tabela, ocorrerão 6 falhas de página após o processo acessar todas as páginas. Logo, o número de falhas não será o mesmo, pois o algoritmo FIFO terá gerado uma falha a mais do que o LRU.

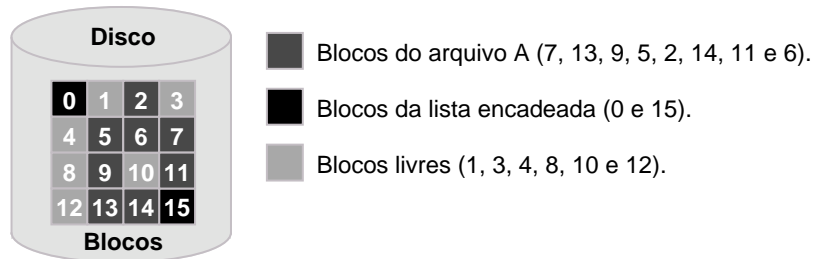
Página	Ordenação		Falha?
2	2		Sim
1	2	1	Sim
2	2	1	Não
3	1	3	Sim
2	3	2	Sim
1	2	1	Sim
3	1	3	Sim
1	1	3	Não

4. (1,5) Suponha que um processo tenha cinco segmentos, com tamanhos de 2MB, 15MB, 105MB, 295MB e 500MB. Se a segmentação com paginação for usada pelo sistema operacional, e se os possíveis tamanhos da página virtual forem 1MB, 8MB, 64MB e 256MB, para quais destes tamanhos poderemos armazenar todos os segmentos em uma memória física de 1024MB? Dentre os tamanhos de página que permitem armazenar todos os segmentos, existe um tamanho para o qual não exista fragmentação interna em todos os segmentos? Justifique a sua resposta.

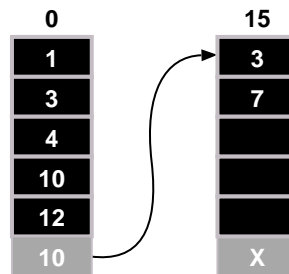
Resp.: Na tabela a seguir, mostramos o número de molduras de página (igual ao número de páginas virtuais), necessárias para armazenar todos os segmentos. Na primeira coluna mostramos o tamanho da moldura. Na segunda coluna mostramos o número total de molduras para o tamanho dado na primeira coluna. Nas colunas 3 a 7 mostramos a quantidade de molduras usadas pelos segmentos do processo. Finalmente, na última coluna, a 8, mostramos o número total de molduras necessárias para armazenar simultaneamente todos os segmentos. Como podemos ver pela tabela, somente podemos armazenar todos os segmentos para os tamanhos de página de 1MB e 8MB. Agora, para que não exista fragmentação interna, os tamanhos de todos os segmentos devem ser múltiplos do tamanho da página. Como os tamanhos são todos múltiplos de 1MB, e como nenhum dos tamanhos é múltiplo de 8MB, então somente não teremos fragmentação interna se o tamanho da página for de 1MB.

Tamanho da moldura	Número de molduras	Segmentos					Total de molduras
		2MB	15MB	105MB	295MB	500MB	
1MB	1024	2	15	105	295	500	917
8MB	128	1	2	14	37	63	117
64MB	16	1	1	2	5	8	17
256MB	4	1	1	1	2	2	7

5. (1,5) Suponha que um disco tenha 16 blocos, numerados de 0 até 15, e que os blocos lógicos do arquivo A, o único armazenado no disco, estejam, em ordem, associados aos blocos físicos dados na figura. Suponha ainda que uma lista encadeada seja usada para gerenciar os blocos livres do disco, sendo que cada bloco pode armazenar até 6 números de blocos. Um aluno de sistemas operacionais afirmou que a lista dada na figura a seguir representa corretamente uma lista com os blocos livres do disco. A lista dada pelo aluno está correta? Se você acha que sim, basta dizer isso mas, se você acha que não, diga os erros existentes na lista.



Lista encadeada (Blocos 0 e 15)



Resp.: A figura do aluno está errada porque existem quatro erros na lista. O primeiro erro é que o bloco 8 está livre mas não está na lista,

logo ele é um bloco **ausente** que não poderá ser usado até a lista ser corrigida. O segundo erro é que o bloco livre 3 está duplicado na lista, o que poderá, no futuro, fazer com que ele seja alocado a dois arquivos diferentes do disco ou até mesmo alocado novamente ao mesmo arquivo. Já o terceiro erro é que o bloco 7, pertencente ao arquivo A, foi marcado como livre, o que poderá fazer com que esse bloco seja alocado a um outro arquivo no futuro ou até mesmo novamente a A. Finalmente, o último erro é que o número do próximo bloco da lista, indicado pelo último número armazenado no bloco 0, deveria ser 15 ao invés de 10, pois 15 é o próximo bloco da lista após o 0.

6. (1,5) Diga a quais conceitos vistos em aula se referem as seguintes definições:

- (a) (0,5) Nome dado ao algoritmo de substituição de páginas que ordena as páginas, em ordem crescente, de acordo com o tempo da sua cópia para a memória, para depois escolher a página copiada há mais tempo para ser substituída.

Resp.: FIFO.

- (b) (0,5) Nome dado ao conjunto composto pelas páginas atualmente acessadas por um dado processo.

Resp.: Conjunto funcional.

- (c) (0,5) Nome dado ao arquivo especial composto por um conjunto de entradas, sendo que cada entrada contém o nome de um arquivo e os seus atributos ou um ponteiro para uma estrutura com os seus atributos.

Resp.: Diretório.