



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 2006

Nome -
Assinatura -

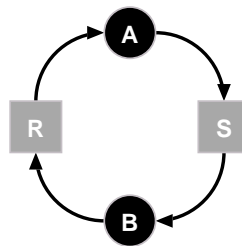
Observações:

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. (1.0) Em relação ao gerenciamento dos dispositivos físicos, descreva sucintamente as funções desempenhadas pelas camadas de Hardware de E/S e de Software de E/S.

Resp.: A camada de Hardware de E/S lida com os dispositivos de E/S, isto é, o funcionamento interno dos dispositivos físicos, e como estes dispositivos podem ser manipulados pelo sistema operacional através de comandos enviados às controladoras que os controlam. Já a camada de Software de E/S são as partes do sistema operacional que tratam do gerenciamento dos dispositivos de E/S do computador. Esta camada é dividida em várias subcamadas. As camadas iniciais tratam do gerenciamento direto dos dispositivos de E/S através de comandos enviados às suas controladoras. As camadas finais disponibilizam, para o resto do sistema, dispositivos abstratos que são mais fáceis de serem usados do que os dispositivos de E/S reais do computador. Por exemplo, quando acessamos um arquivo no disco, a camada de Hardware de E/S trata de inicializar o disco, de procurar neste onde está o arquivo, e de começar a enviar o arquivo. Já a camada de Software de E/S possui as funções que gerenciam o sistema de arquivos, e também trata do envio à controladora do disco dos comandos necessários para iniciar a leitura do arquivo.

2. (2.1) Na figura dada a seguir mostramos um grafo de recursos para dois processos A e B , e dois recursos R e S . Em quais das situações dadas a seguir teremos um impasse? Justifique a sua resposta.



- (a) (0.7) R e S são recursos preemptivos.

Resp.: Não teremos um impasse, pois como tanto R como S são preemptivos, basta tirarmos o recurso de um dos processos e

dá-lo ao outro processo. Depois, quando o processo que recebeu os recursos os liberar, passamos ambos os recursos para o outro processo. Por exemplo, poderíamos tirar o recurso R do processo A e dá-lo ao processo B , que agora poderia reiniciar a sua execução. Após B liberar os recursos R e S , poderíamos finalmente alocar estes recursos para A , que poderia então continuar com a sua execução.

(b) (0.7) R e S são recursos não-preemptivos.

Resp.: Neste caso teremos um impasse, pois como ambos os recursos são não-preemptivos, somente o processo A pode liberar R e o processo B pode liberar S , sem que isso prejudique os resultados das computações que estão sendo executadas por eles.

(c) (0.7) R é um recurso preemptivo e S é um recurso não-preemptivo.

Resp.: Não, pois como R é um recurso preemptivo, podemos tirá-lo do processo A e alocá-lo ao processo B . Quando B liberar ambos os recursos, poderemos retornar o recurso R para A e alocar o recurso S para ele, o que permitirá que A continue normalmente a sua execução.

3. (2.0) Suponha que um computador possui quatro molduras de página e que o seu espaço de endereçamento possui oito páginas virtuais. Quantas falhas de página ocorrerão ao acessarmos, em seqüência, as páginas 0, 3, 4, 1, 2, 3, 7 e 5, usando cada um dos algoritmos dados a seguir e supondo que as molduras estão inicialmente vazias?

(a) (1.0) Algoritmo de segunda chance, sendo que o bit R se torna 1 ao acessarmos a página, e somente é zerado pelo algoritmo.

Resp.: Se usarmos o algoritmo de segunda chance teremos um total de 7 falhas de página, como poderemos ver pela tabela dada a seguir. A ordem em que as páginas estão na fila FIFO é dada na segunda coluna desta tabela, sendo que ao lado de cada página colocamos, entre parênteses, o valor do seu bit R . Para cada página, acessada na ordem dada pela questão, mostramos na tabela o que

ocorre com a fila ao acessá-la. Note que 3 páginas, a 0, a 4 e a 1, foram substituídas e salvas no disco pelo algoritmo de segunda chance.

Página referenciada	Conteúdo das molduras de página				Ocorreu uma falha?
0	0 (1)				Sim
3	0 (1)	3 (1)			Sim
4	0 (1)	3 (1)	4 (1)		Sim
1	0 (1)	3 (1)	4 (1)	1 (1)	Sim
2	3 (1)	4 (1)	1 (1)	0 (0)	Sim
	4 (1)	1 (1)	0 (0)	3 (0)	
	1 (1)	0 (0)	3 (0)	4 (0)	
	0 (0)	3 (0)	4 (0)	1 (0)	
	3 (0)	4 (0)	1 (0)	2 (1)	
3	3 (1)	4 (0)	1 (0)	2 (1)	Não
7	4 (0)	1 (0)	2 (1)	3 (0)	Sim
	1 (0)	2 (1)	3 (0)	7 (1)	
5	2 (1)	3 (0)	7 (1)	5 (1)	Sim

(b) (1.0) Algoritmo LRU.

Resp.: Se usarmos o algoritmo LRU também teremos 7 falhas de página. Na tabela dada a seguir damos, na segunda coluna, a ordem das páginas segundo a última vez em que foram acessadas. Cada linha da tabela representa o acesso a uma página na ordem dada pela questão. Note que as páginas 0, 4 e 1 foram escolhidas, substituídas, e salvas no disco pelo algoritmo LRU.

Página referenciada	Conteúdo das molduras de página				Ocorreu uma falha?
0	0				Sim
3	0	3			Sim
4	0	3	4		Sim
1	0	3	4	1	Sim
2	3	4	1	2	Sim
3	4	1	2	3	Não
7	1	2	3	7	Sim
5	2	3	7	5	Sim

4. (2.0) Qual é a diferença entre a técnica de segmentação e a de paginação? É possível usarmos ao mesmo tempo estas duas técnicas? Justifique a sua resposta.

Resp.:Na técnica de segmentação, o sistema operacional disponibiliza, para cada processo, um espaço de endereçamento bidimensional. Cada endereço virtual é composto por duas partes, o **segmento** e o **endereço** dentro deste segmento. Um segmento define um espaço de endereçamento linear separado, com endereços variando de 0 até um endereço máximo, igual ao seu tamanho menos um. O tamanho de um segmento não pode ser maior do que o tamanho da memória física. Como o programador está ciente da existência dos segmentos, ele pode definir mecanismos de proteção, como os acessos para leitura, escrita e execução. O problema com a técnica de segmentação é, além do limite do tamanho do segmento, o da fragmentação externa, que é gerada com a contínua movimentação dos segmentos dos processos entre a memória e o disco. Já na técnica de paginação, o sistema operacional disponibiliza um único espaço de endereçamento virtual para cada processo, dividido uniformemente em um conjunto de regiões de memória chamadas de **páginas virtuais**. O programador não sabe da existência destas páginas, e com isso, não pode definir mecanismos de proteção para restringir certos tipos de acesso a algumas delas (somente o sistema operacional pode fazer isso, mas este o faz sem o conhecimento do programador). Ao contrário da segmentação, o tamanho do espaço

de endereçamento virtual pode ser maior do que o do espaço de endereçamento físico, pois a memória física pode ser dividida em um conjunto de regiões, chamadas de **molduras de página**, com mesmo tamanho das páginas virtuais. Neste caso, cada página será mapeada, quando for necessário, para alguma moldura da memória física. O problema com a técnica de paginação é, como vimos na Aula 10, o da fragmentação interna, pois em média metade do espaço da última página é desperdiçado.

-Sim, a segmentação pode ser usada com a paginação, pois como cada segmento é um espaço de endereçamento linear, ele poderá ser paginado. Neste caso poderemos ter segmentos com tamanhos maiores do que o da memória física, e, quando for necessário, cada página virtual acessada do segmento poderá ser mapeada para uma moldura de página da memória física.

5. (2.0) Suponha que desejamos acessar, nesta ordem, as posições 5, 7, 2, 1 e 8 do arquivo teste.c dado abaixo. Quais posições deveremos necessariamente acessar, se usarmos:

- (a) (1.0) o acesso seqüencial?

Resp.: Primeiramente lembremos que se estivermos em uma posição x do arquivo e desejarmos acessar uma posição $y < x$, deveremos executar uma operação de **rewind** no arquivo e acessar as posições de 1 até y dele, em seqüência. Porém, se $y \geq x$, deveremos acessar as posições de x até y dele. Logo, ao acessar a posição 5, deveremos acessar as posições 1, 2, 3, 4 e 5. Para acessar a posição 7, temos que acessar as posições 6 e 7. Agora, para acessar a posição 2, deveremos executar a operação **rewind** e acessar as posições 1 e 2. Depois, para acessar a posição 1, temos que executar mais uma vez a operação **rewind** e acessar a posição 1. Finalmente, para acessar a posição 8, deveremos acessar, em seqüência, as posições 2, 3, 4, 5, 6, 7 e 8. Com isso, acessaremos todas as posições do arquivo ao usarmos o acesso seqüencial.

- (b) (1.0) o acesso aleatório?

Resp.: No acesso aleatório podemos acessar qualquer posição x

do arquivo imediatamente, bastando usar a operação **seek** para saltar para esta posição x . Logo, podemos acessar cada posição na ordem dada, isto é, 5, 7, 2, 1 e 8, simplesmente executando, antes de acessar cada posição, uma operação **seek** para saltar para ela, para depois então acessá-la. Com isso, as posições do arquivo acessadas, ao usarmos o acesso aleatório, serão a 1, a 2, a 5, a 7 e a 8.

Arquivo teste.c (8 posições)

1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---

6. (1.0) Qual é o objetivo de usarmos os mecanismos de proteção em um sistema de arquivos?

Resp.: Alguns arquivos de um sistema de arquivos podem armazenar informações críticas, que devem ser acessíveis somente a alguns usuários do sistema, sendo que o tipo de acesso (em geral, leitura, escrita e execução) pode variar para cada um destes usuários. O objetivo do uso dos mecanismos de proteção é exatamente o de definir, para cada arquivo do sistema, quais usuários podem acessá-lo, e quais operações cada um deles pode executar sobre os arquivos. O sistema operacional deve garantir, para cada arquivo, que somente os usuários autorizados irão acessá-lo, e que para cada um destes usuários somente sejam executadas as operações permitidas para ele sobre o arquivo.