



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
AP3 - Segundo Semestre de 2010

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. (2,0) 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) (0,4) Um sistema multiprogramado pode ser monousuário ou multiusuário.

Resp.: V (Verdadeira).

- (b) (0,4) Uma propriedade essencial de qualquer processo é que o sistema operacional deve ser capaz de pará-lo e reiniciá-lo mais tarde sem que isso afete o resultado de sua execução.

Resp.: V (Verdadeira).

- (c) (0,4) O interpretador de comandos é a principal interface entre um processo e o sistema operacional, pois é o responsável por executar as chamadas ao sistema operacional.

Resp.: F (Falsa, pois são as chamadas ao sistema operacional que são a principal interface entre o sistema operacional e os processos. O interpretador de comandos é a principal interface entre um usuário e o sistema operacional).

- (d) (0,4) Quando a multiprogramação é gerenciada pelo sistema operacional, ela não é visível ao usuário. Quando ela é gerenciada por uma extensão do sistema operacional, ela sempre é visível ao usuário.

Resp.: F (Falsa, pois é exatamente o contrário, isto é, a multiprogramação é visível ao usuário quando é gerenciada pelo sistema operacional, e não é visível ao usuário quando ela é gerenciada por uma extensão do sistema operacional).

- (e) (0,4) Quando o núcleo do sistema operacional é dividido em camadas, cada uma fornece para as camadas superiores uma versão

abstrata da parte do hardware gerenciada por ela.

Resp.: V (Verdadeira).

2. (2,0) Quais são as quatro condições necessárias para a ocorrência de um impasse?

Resp.: As quatro condições necessárias para a ocorrência de um impasse são as seguintes: (i) condição de exclusão mútua (um recurso está associado a um único processo por um dado intervalo de tempo); (ii) condição de segura e espera (um processo pode solicitar novos recursos mesmo que já possua alguns recursos); (iii) condição de nenhuma pre-empção (somente um processo pode liberar um recurso dado a ele); e (iv) condição de espera circular (existe uma cadeia circular de espera por recursos entre um conjunto de processos).

3. (2,0) Qual é a diferença, em relação ao gasto de tempo pelo processador, entre usar apenas uma instrução TSL e usar um semáforo para garantir a exclusão mútua?

Resp.: A diferença tem a ver com o desperdício de tempo do processador ao esperar por um evento externo. Como vimos na Aula 5, a instrução TSL precisa usar uma posição de memória LOCK e um registrador REGISTER do processador. Como a função ENTER_REGION, também dada na Aula 5, fica continuamente verificando se REGISTER é diferente de 0 (note que a instrução TSL altera o valor de REGISTER), então o processo não é bloqueado, e há um desperdício do tempo do processador devido a esta verificação. O desperdício somente acabará quando um outro processo deixar a sua seção crítica, e depois chamar LEAVE_REGION (dada também na Aula 5). Agora, quando um semáforo binário é usado, o processo é bloqueado ao executar a operação **P** sobre ele caso um outro processo esteja na sua seção crítica. O processo somente será desbloqueado quando o outro processo sair da sua seção crítica e depois executar a operação **V** sobre o semáforo. Como o processo foi bloqueado, não há mais desperdício do tempo do

processador enquanto espera para executar a sua seção crítica.

4. (2,0) Suponha que o sistema operacional tenha alocado três molduras de página inicialmente vazias a um processo, e que o processo tenha acessado, nesta ordem, as páginas 1, 3, 4, 3, 4, 7 e 3. Indique o conjunto de páginas residentes nas molduras após cada um desses acessos quando o algoritmo de substituição em uso é o LRU.

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 será a primeira página segundo esta ordenação, isto é, a página não acessada há mais tempo. Na figura dada a seguir mostramos, em cada linha, o que ocorre ao acessarmos as páginas na ordem dada no enunciado. Para cada uma destas linhas mostramos, na primeira coluna a página que é acessada, na segunda coluna a ordem em que as páginas devem ser escolhidas e, finalmente, na terceira coluna se o acesso gera uma falha de página.

Páginas	Ordenação	Ocorreu uma falha?
1	1	Sim
3	1 3	Sim
4	1 3 4	Sim
3	1 4 3	Não
4	1 3 4	Não
7	3 4 7	Sim
3	4 7 3	Não

5. (2,0) Por que o acesso aos arquivos do disco será, em geral, mais eficiente, se o sistema operacional usar a alocação contígua para gerenciar os blocos do disco alocados aos arquivos?

Resp.: Como vimos na Aula 11, na alocação contígua, para cada arquivo os blocos lógicos são armazenados, em ordem, em blocos físicos consecutivos do disco. Logo, o acesso a cada arquivo será, em geral,

mais eficiente porque, como os blocos do arquivo não estão dispersos aleatoriamente no disco, a controladora precisará fazer menos operações de busca de setores e de posicionamento das cabeças do disco. Note que estas operações são bem mais lentas do que a operação de leitura de um bloco.