

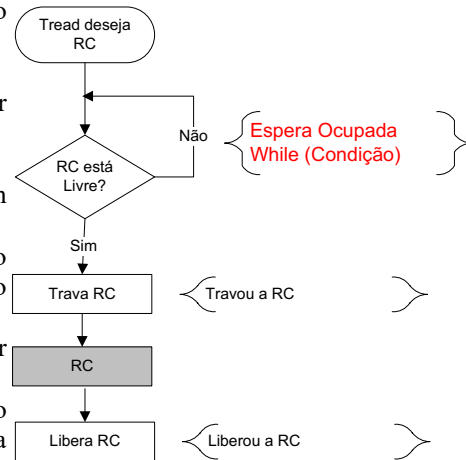
Aluno(a) : **Gabário**

(29) **1ª Questão :** O problema da “espera ocupada” é gerado pela execução de loop indefinido enquanto a variável de semáforo impede a entrada na região crítica. Para resolver este problema, é utilizado o algoritmo de *sleep / wake-up*. Coloque ao lado a soma das afirmativas que tratam corretamente deste processo:

- ✓ (01) Diz-se que o método de *sleep / wake-up* aumenta a performance do sistema por bloquear o processo enquanto o semáforo estiver fechado, liberando o processador para o atendimento de outros processos;
- (02) Embora seja bastante eficiente, o bloqueio de um processo que tenta acessar a região crítica com o semáforo fechado pode provocar instabilidade do sistema, devido ao risco de *dead-lock* derivado do bloqueio;
- ✓ (04) Chama-se de *wake-up* o desbloqueio do processo, que volta para fila de pronto;
- ✓ (08) Vários processos podem ser “acordados”, mas invariavelmente apenas um terá acesso à região crítica, já que todos os demais que tentarem acesso à área crítica serão novamente adormecidos;
- ✓ (16) Adormecer, neste caso, é sinônimo de “ser bloqueado”.

(53) **2ª Questão :** Dado o algoritmo ao lado, preencha o espaço com a soma das alternativas corretas:

- ✓ (01) A região crítica só pode ser executada por um processo por vez;
- (02) Um processo na RC pode ser bloqueado por I/O;
- ✓ (04) Preempções entre o teste e o travamento da RC podem provocar o acesso de múltiplos processos à Região Crítica;
- (08) O uso de uma instrução específica *Test-And-Set* permite o acesso simultâneo de dois ou mais processos à Região Crítica;
- ✓ (16) A variável que controla o acesso à Região Crítica deve ter escopo global;
- ✓ (32) Desabilitar interrupções pode impedir o acesso simultâneo à RC, porém implica em problemas de outra ordem, como a dificuldade na atualização do *clock*, por exemplo.



3ª. Questão : Compare os métodos do extermínio e da volta-ao-passado quanto à eficiência e desempenho na eliminação de *dead-locks*:

4ª. Questão : Para cada uma das afirmações abaixo, assinale (C)erto ou (E)rrado. No caso da opção estar errada, assinale também logo abaixo a(s) justificativa(s) para o erro. Se nenhuma das justificativas for adequada, escreva no último espaço a sua justificativa.

Observação : Nesta questão, só será considerada correta a questão marcada como (C)erta e realmente certa ou a questão (E)rrada com a respectiva justificativa marcada. Todos os outros casos serão considerados erros.

- a) (**C**) Existem 4 condições que, se pelo menos uma delas for evitada, não é possível a ocorrência de *dead-locks*: exclusão mútua, posse e espera, não-preempção, espera circular.

Justificativa caso esteja errada:

- () Na verdade, TODAS as condições precisam ser evitadas;
() Essas não são as únicas condições necessárias;
() A ocorrência de qualquer uma destas condições na verdade PROVOCA o *dead-lock*;
() _____

- b) (**E**) O uso de uma instrução TEST-AND-SET, embora seja um método interessante para garantir a exclusão mútua, pode falhar caso ocorra preempção entre o teste e a gravação da variável de semáforo.

Justificativa caso esteja errada:

- (**X**) Na verdade, o objetivo da instrução TEST-AND-SET é justamente evitar a ocorrência deste problema.
() A preempção não evita a exclusão mútua, já que, ao retornar, o processo estará exatamente na mesma instrução.
() A exclusão mútua não pode ser garantida por apenas este mecanismo.
() _____

- c) (**C**) No gerenciamento de memória, um endereço lógico só é equivalente ao físico quando a aplicação é carregada na memória a partir do endereço inicial.

Justificativa caso esteja errada:

- () Um endereço lógico não pode ser igual ao físico, já que precisa ser convertido durante o processo de carga da aplicação.
() Um endereço lógico sempre se equivale ao físico após a carga da aplicação na memória do computador.
() Na verdade, os endereços ficam iguais quando o processo é carregado nas últimas posições de memória.
() _____

5ª. Questão : Relacione, usando suas próprias palavras, quais as principais diferenças entre a alocação de memória em partições fixas, e em partições variáveis. Apresente as diferenças em termos operacionais, e também as vantagens e desvantagens dos dois modelos:

A divisão em partições de tamanho fixo tem implementação mais simples, o que aumenta o desempenho no processo de alocação e liberação de blocos de memória. Por outro lado, devido à falta de flexibilidade na divisão do espaço disponível, pode haver subutilização de memória. O sistema de partições variáveis tem propriedades opostas.

6ª Questão : Com base no seu conhecimento sobre a alocação de memória nos Sistemas Operacionais, avalie as afirmativas abaixo:

- I. Todo processo em execução precisa estar completamente armazenado na memória principal para poder ser executado.
- II. Uma determinada instrução a ser executada precisa estar armazenada na memória principal do computador.
- III. Diz-se que uma aplicação foi devidamente compilada quando pode ser carregada diretamente, sem modificações, na memória do computador.

- ☒ Apenas a segunda afirmativa está correta;
(b) As duas primeiras afirmativas estão corretas;
(c) Todas as afirmativas estão corretas;
(d) Nenhuma das afirmativas está correta.

7ª Questão : Considere que um endereço gerado pela CPU é chamado de endereço lógico, enquanto um endereço visto pela unidade de memória, ou seja, aquele carregado no registrador de endereço da memória, é normalmente chamado de endereço físico.

Com base nisso, assinale (C)erto ou (E)rrado nas afirmativas abaixo. No caso da opção estar errada, assinale também logo abaixo a(s) justificativa(s) para o erro. Se nenhuma das justificativas for adequada, escreva no último espaço a sua justificativa.

Observação : Nesta questão, só será considerada correta a questão marcada como (C)erta e realmente certa ou a questão (E)rrada com a respectiva justificativa marcada. Todos os outros casos serão considerados erros.

- a) (☒) O programa de usuário pode nunca ver os endereços físicos reais. O programa pode criar um ponteiro para a posição lógica 346, armazená-la na memória, manipulá-la e compará-la com outros endereços; tudo isso como o número 346. Somente quando ela for usada como um endereço de memória (em uma carga ou armazenamento indireto, por exemplo) ela será realocada em relação ao registrador de base.

Justificativa caso esteja errada:

- () Ao executar um programa de usuário, os endereços são primeiramente convertidos antes do armazenamento do código na memória principal;
() O registrador de base não realoca os endereços de memória. Ele apenas determina o local onde deve ser armazenado o código da aplicação a ser executada;
() Em toda e qualquer operação de memória sempre é necessária a conversão do endereço lógico para o endereço físico;
() _____

- b) (☒) Nem sempre um processo inteiro precisa estar na memória física para ser executado. Na carga dinâmica, por exemplo, apenas a parte em execução do código precisa estar carregada na memória. Todas as demais partes do código podem ficar armazenadas na memória secundária, tipicamente em formato relocável.

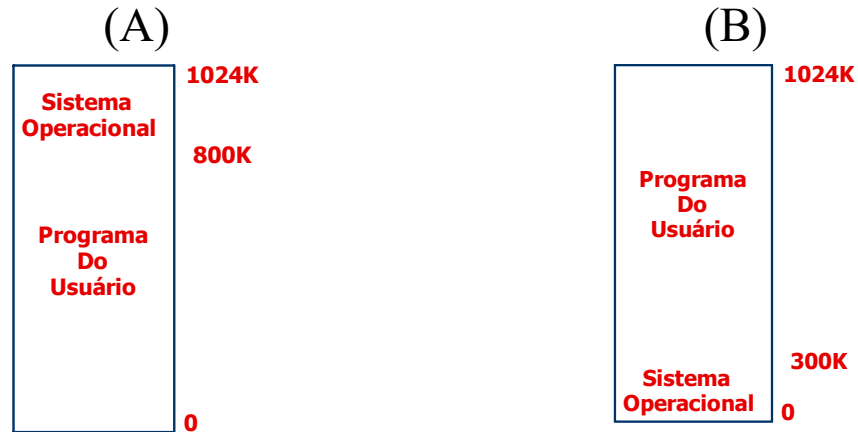
Justificativa caso esteja errada:

- () Não só a parte em execução precisa estar carregada, como também a parte em execução anterior, e também a provável próxima rotina a ser executada;
() Todas as partes do programa em execução precisam estar carregadas na memória principal para serem executadas;
() A realocação só é feita em tempo de execução. Assim não é necessário manter as outras partes do programa em formato relocável;
() _____

8ª. Questão : Apresente as diferenças entre as estratégias *Worst-fit*, *Best-fit* e *First-fit* na alocação de um bloco de memória livre do conjunto de blocos disponíveis:

No modo Worst-Fit, a aplicação é carregada, entre os blocos livres de memória que têm espaço suficiente para carregar a aplicação, no maior deles. No Best-Fit, é escolhido o menor bloco que tem espaço disponível para a aplicação. No First-Fit, a aplicação é carregada no primeiro bloco encontrado que tenha espaço suficiente para a aplicação.

(13) **9ª Questão :** No modelo de **alocação contígua fixa**, tipicamente utilizado em sistemas operacionais monotarefa, o sistema operacional podia ser armazenado na parte alta (A) ou na parte baixa (B) da memória principal do computador. Com base nas diferenças entre as duas estratégias, escreva ao lado a soma dos números das afirmativas corretas:



- ✓ (01) Em ambas as estratégias, o programa do usuário fica limitado à capacidade máxima de memória disponível;
- (02) Na estratégia onde o programa do usuário é carregado após o Sistema Operacional (B), o endereço físico é igual ao endereço lógico;
- ✓ (04) Em ambas as estratégias, um registrador de controle delimita a área de memória reservada ao sistema operacional, impedindo o acesso indevido pela aplicação do usuário;
- ✓ (08) Para garantir a execução de processos maiores, o programa é dividido em módulos, que podem inclusive ser carregados na medida em que são necessários, ocupando sempre a mesma área de memória (técnica de *overlays*);
- (16) Através de processo de configuração, a área disponível é dividida em partes de tamanhos pré-definidos, que são estabelecidos durante o *boot* do sistema operacional.

(17) **10ª Questão :** Sobre o conceito de proteção de memória, escreva ao lado a soma dos números das afirmativas corretas:

- ✓ (01) Se um processo em execução na partição 1 (entre os endereços 120K e 200K) tentar acessar a posição 80.550, o acesso será impedido, pois trata-se de área do S.O.(entre os endereços 0K e 120K);
- (02) Os programas em execução nas partições 1, 2 e 3 (que não contém o SO) têm acesso uns às áreas específicas dos outros, sem qualquer problema.
- (04) Apenas a partição 3 (com endereços começando em 300K), que opera em um endereço físico igual ao endereço Lógico, deve ter o acesso à sua respectiva área controlado;
- (08) Apenas a troca da posição de memória utilizada para carga do SO pode promover a devida proteção de memória (o SO deve ser colocado no topo da pilha de memória);
- ✓ (16) O registrador de relocação e o registrador de limite determinam os limites inferior e superior da partição de memória que será devidamente protegida.

Memória	
0	S.O.
120K	
200K	Partição 1
300K	Partição 2
450K	Partição 3