

## LISTA DE EXERCÍCIOS - RESPOSTAS

1. Por que o espaço de endereço de um processo é dividido em várias regiões?
2. (V/F) Os termos 'processo' e 'programa' são sinônimos.

**Respostas:** 1) Cada região de um espaço de endereço comumente contém informações que são acessadas de um modo similar. Por exemplo, a maioria dos processos lê e executa instruções, mas não modifica suas instruções. Processos lêem e escrevem de e para a pilha, mas na ordem 'último a entrar, primeiro a sair'. Processos lêem e escrevem dados em qualquer ordem. Separar o espaço de endereço de um processo em diferentes regiões habilita o sistema operacional a impor essas regras de acesso. 2) Falso. Um processo é um programa em execução; um programa é uma entidade inanimada.

1. (V/F) A qualquer dado instante somente um processo pode executar instruções em um computador.
2. Um processo entra no estado de *bloqueado* quando está esperando que um evento ocorra. Cite diversos eventos que podem fazer um processo entrar em estado de *bloqueado*.

**Respostas:** 1) Falso. Um computador multiprocessador pode ter tantos processos em execução quantos são os processadores. 2) Um processo pode entrar em estado de bloqueado se emitir uma requisição de dados localizados em um dispositivo de latência alta, tal como um disco rígido, ou requisições de um recurso que está alocado a um outro processo e indisponível naquele momento (por exemplo, uma impressora). Um processo também pode ficar bloqueado até que ocorra um evento, como a utilização de teclado ou a movimentação do mouse pelo usuário.

1. Como o sistema operacional impede que um processo monopolize um processador?
2. Qual a diferença entre processos que estão acordados e processos que estão adormecidos?

**Respostas:** 1) Um relógio de interrupção gera uma interrupção após um quantum de tempo especificado, e o sistema operacional despacha um outro processo para executar. O processo interrompido executará novamente quando alcançar o topo da lista de prontos e um processador ficar disponível novamente. 2) Um processo acordado está disputando ativamente um processador; um processo adormecido não pode usar um processador mesmo que haja um disponível.

1. Qual a finalidade da tabela de processos?
2. (V/F) A estrutura de um PCB depende da implementação do sistema operacional.

**Respostas:** 1) A tabela de processos habilita o sistema operacional a localizar o PCB de cada processo. 2) Verdadeiro.

1. (V/F) Um processo pode ter um número zero de processos-pai.
2. Por que é vantajoso criar uma hierarquia de processos em vez de uma lista encadeada?

**Respostas:** 1) Verdadeiro. O primeiro processo criado, geralmente denominado *init* em sistemas UNIX, não tem um pai. E também, em alguns sistemas, quando um processo-pai é destruído, seus filhos prosseguem independentes, sem seu pai. 2) Uma hierarquia de processos permite que o sistema operacional controle os relacionamentos pai/filho entre processos, o que simplifica operações como localizar todos os processos-filho de um determinado processo-pai quando esse for extinto.

1. Quais são as três maneiras pelas quais um processo pode chegar ao estado *suspense-pronto*?
2. Em qual cenário é melhor suspender um processo em vez de abortá-lo?

**Respostas:** 1) Um processo pode chegar ao estado *suspense-pronto* se for suspenso do estado *em execução*; se for suspenso do estado *pronto* por um processo *em execução*; ou se estiver no estado *suspense-bloqueado* e ocorrer a conclusão da E/S ou do evento pelo qual está esperando. 2) Quando um usuário ou administrador de sistema suspeitar do comportamento de um processo, mas não quiser perder o trabalho realizado pelo processo, é melhor suspender o processo para poder examiná-lo.

1. A partir de onde um sistema operacional carrega o contexto de execução para o processo a ser despachado durante um chaveamento de contexto?
2. Por que um sistema operacional deve minimizar o tempo requerido para realizar um chaveamento de contexto?

*Respostas:* 1) As informações de contexto do processo a ser despachado são armazenadas em seu PCB. 2) Durante um chaveamento de contexto um processador não pode executar instruções em nome de processos, o que pode reduzir o rendimento.

1. O que significa interrupção síncrona?
2. Cite uma alternativa para interrupção e explique por que raramente é usada.

*Respostas:* 1) Uma interrupção síncrona ocorre devido à execução de software. 2) Um sistema pode executar sondagem, na qual um processador verifica periodicamente o estado dos dispositivos. Essa técnica raramente é usada, porque cria significativa sobrecarga quando o processador sonda dispositivos cujo status não mudou. Interrupções eliminam essa sobrecarga notificando um processador somente quando o status de um dispositivo mudar.

1. Por que as localizações dos tratadores de interrupção geralmente não são armazenadas em uma lista encadeada?
2. Por que o contexto de execução do processo é salvo na memória enquanto o tratador de interrupção executa?

*Respostas:* 1) Para evitar que fique assobrecado por interrupções, o sistema precisa ser capaz de processar cada interrupção rapidamente. Passar por uma lista encadeada poderia aumentar significativamente o tempo de resposta de um sistema, caso o número de tipos de interrupção fosse grande. Portanto, muitos dos sistemas usam um vetor de interrupção

(um array) para acessar rapidamente a localização de um tratador de interrupção. 2) Se o contexto de execução do processo não for salvo na memória, o tratador de interrupção poderá escrever sobre os registradores do processo.