

## Questões e respostas do capítulo 1 do livro Sistemas Operacionais: Conceitos e Mecanismos

1. Quais os dois principais objetivos dos sistemas operacionais?

R: Abstração de recursos e Gerência de recursos.

2. Por que a abstração de recursos é importante para os desenvolvedores de aplicações? Ela tem utilidade para os desenvolvedores do próprio sistema operacional?

R: A abstração de recursos simplifica a construção de programas aplicativos, torna os aplicativos independentes do hardware e através de suas abstrações, o sistema operacional permite aos aplicativos usar a mesma interface para dispositivos diversos.

3. A gerência de atividades permite compartilhar o processador, executando mais de uma aplicação ao mesmo tempo. Identifique as principais vantagens trazidas por essa funcionalidade e os desafios a resolver para implementá-la.

R: O uso desses processadores deve ser distribuído entre os aplicativos presentes no sistema, de forma que cada um deles possa executar na velocidade adequada para cumprir suas funções sem prejudicar os demais. Entretanto, em um sistema com várias atividades simultâneas, podem surgir conflitos no uso do hardware, quando dois ou mais aplicativos precisam dos mesmos recursos para poder executar. Cabe ao sistema operacional definir políticas para gerenciar o uso dos recursos de hardware pelos aplicativos, e resolver eventuais disputas e conflitos.

4. O que caracteriza um sistema operacional de tempo real? Quais as duas classificações de sistemas operacionais de tempo real e suas diferenças?

R: Um sistema operacional de tempo real tem um comportamento temporal previsível (ou seja, seu tempo de resposta deve ser conhecido no melhor e pior caso de operação). A estrutura interna de um sistema operacional de tempo real deve ser construída de forma a minimizar esperas e latências imprevisíveis, como tempos de acesso a disco e sincronizações excessivas. Existem duas classificações de sistemas de tempo real: soft real-time systems, nos quais a perda de prazos implica na degradação do serviço prestado. Um exemplo seria o suporte à gravação de CDs ou à reprodução de músicas.

5. O que diferencia o núcleo do restante do sistema operacional?

R: O núcleo é o coração do sistema operacional, responsável pela gerência dos recursos do hardware usados pelas aplicações. Ele também implementa as principais abstrações utilizadas pelos programas aplicativos.

6. Seria possível construir um sistema operacional seguro usando um processador que não tenha níveis de privilégio? Por quê?

R: Não, para assegurar a integridade do sistema operacional, é essencial garantir que as aplicações não consigam acessar o hardware diretamente, mas sempre através de pedidos ao sistema operacional, que avalia e intermedeia todos os acessos ao hardware

7. O processador Pentium possui dois bits para definir o nível de privilégio, resultando em 4 níveis distintos. A maioria dos sistemas operacionais para esse processador usa somente os níveis extremos (0 e 3, ou  $00_2$  e  $11_2$ ). Haveria alguma utilidade para os níveis intermediários?

R: Sim, se com divisão de processamento entre os níveis podem ser usados para oferecer uma política de segurança

8. Quais as diferenças entre interrupções, exceções e traps?

R: As *interrupções* são eventos causados por dispositivos externos ao processador, *exceções* são eventos gerados pelo próprio processador, *traps* são eventos causados por software.

9. Quais as implicações de mascarar interrupções? O que pode ocorrer se o processador ignorar interrupções por muito tempo? O que poderia ser feito para evitar o mascaramento de interrupções?

R: Cada interrupção geralmente corresponde a um evento ocorrido em um dispositivo periférico: a chegada de um pacote de rede, um click no mouse, uma operação concluída pelo controlador de disco, etc. Isso representa centenas ou mesmo milhares de interrupções recebidas por segundo, dependendo da carga e da configuração do sistema (número e natureza dos periféricos). Por isso, as rotinas de tratamento de interrupção devem ser curtas e realizar suas tarefas rapidamente (para não prejudicar o desempenho do sistema). Para distinguir interrupções geradas por dispositivos distintos, cada interrupção é identificada por um inteiro, normalmente com 8 bits. Como cada interrupção pode exigir um tipo de tratamento diferente (pois os dispositivos são diferentes), cada IRQ deve disparar sua própria rotina de tratamento de interrupção.

10. O comando em linguagem C `fopen` é uma chamada de sistema ou uma função de biblioteca? Por quê?

R: É uma função da biblioteca padrão da linguagem C, ela abre o arquivo cujo nome é indicado por filename.

11. Monte uma tabela com os benefícios e deficiências mais significativos das principais arquiteturas de sistemas operacionais.

R: Em um sistema monolítico, todos os componentes do núcleo operam em modo núcleo e se inter-relacionam conforme suas necessidades, sem restrições de acesso entre si, pois o código no nível núcleo tem acesso pleno a todos os recursos e áreas de memória. A grande vantagem dessa arquitetura é seu desempenho: qualquer componente do núcleo pode acessar os demais componentes, toda a memória ou mesmo dispositivos periféricos diretamente, pois não há barreiras impedindo esse acesso. A interação direta entre componentes também leva a sistemas mais compactos. Todavia, a arquitetura monolítica pode pagar um preço elevado por seu desempenho: a robustez e a facilidade de desenvolvimento. Caso um componente do núcleo perca o controle devido a algum erro, esse problema pode se alastrar rapidamente por todo o núcleo, levando o sistema ao colapso (travamento, reinicialização ou funcionamento. Uma forma mais elegante de estruturar um sistema operacional faz uso da noção de camadas: a camada mais baixa realiza a interface com o hardware, enquanto as camadas intermediárias proveem níveis de abstração e gerência cada vez mais sofisticados. Por fim, a camada superior define a interface do núcleo para as aplicações (as chamadas de sistema). Essa abordagem de estruturação de software fez muito sucesso no domínio das redes de computadores, através do modelo de referência OSI (Open Systems Interconnection) [Day, 1983], e também seria de se esperar sua adoção no domínio dos sistemas operacionais. No entanto, alguns inconvenientes limitam sua aceitação nesse contexto: • O empilhamento de várias camadas de software faz com que cada pedido de uma aplicação demore mais tempo para chegar até o dispositivo

periférico ou recurso a ser acessado, prejudicando o desempenho do sistema. • Não é óbvio como dividir as funcionalidades de um núcleo de sistema operacional em camadas horizontais de abstração crescente, pois essas funcionalidades são interdependentes, embora tratem muitas vezes de recursos distintos. Em decorrência desses inconvenientes, a estruturação em camadas é apenas parcialmente adotada hoje em dia. Muitos sistemas implementam uma camada inferior de abstração do hardware para interagir com os dispositivos (a camada HAL – Hardware Abstraction Layer, implementada no Windows NT e seus sucessores), e também organizam em camadas alguns subsistemas como a gerência de arquivos e o suporte de rede (seguindo o modelo OSI). Como exemplos de sistemas fortemente estruturados em camadas podem ser citados o IBM OS/2 e o MULTICS [Corbató and Vyssotsky, 1965]

12. Relacione as afirmações aos respectivos tipos de sistemas operacionais: distribuído (D), multi-usuário (M), desktop (K), servidor (S), embarcado (E) ou de tempo-real (T):

R: [ T ]

[ S ]

[ E ]

[ D ]

[ M ]

[ E ]

[ K ]

[ S ]

[ D ]

[ E ]

13. A operação em modo usuário permite ao processador executar somente parte das instruções disponíveis em seu conjunto de instruções. Quais das seguintes operações não deveriam ser permitidas em nível usuário? Por quê?

R: C - No nível usuário o hardware restringe o uso da memória, permitindo o acesso somente a áreas previamente definidas.

14. Considerando um processo em um sistema operacional com proteção de memória entre o núcleo e as aplicações, indique quais das seguintes ações do processo teriam de ser realizadas através de chamadas de sistema, justificando suas respostas:

R: D - Os sistemas operacionais definem chamadas de sistema para todas as operações envolvendo o acesso a recursos de baixo nível (periféricos, arquivos, alocação de memória, etc.)

15. Coloque na ordem correta as ações abaixo, que ocorrem durante a execução da função `printf("Hello world")` por um processo (observe que nem todas as ações indicadas fazem parte da sequência).

R: [ 5 ]

[ 9 ]

[ 1 ]

[ 4 ]

[ 8 ]

[ 2 ]

[ 7 ]

[ 3 ]

[ 6 ]

16. Considere as afirmações a seguir, relativas aos diversos tipos de sistemas operacionais:

R: As afirmações III e IV estão erradas. III está errado pois é uma característica de sistemas distribuídos e não de rede; e IV está errado pois é uma característica de sistemas desktop e não de tempo real.

17. Considere as afirmações a seguir, relativas às diversas arquiteturas de sistemas operacionais:

R: As afirmações II e V estão corretas. I está errada pois uma máquina virtual de sistema é construída para suportar sistemas operacionais convidados completos; III está errada pois isso ocorre em sistemas monolíticos e não micronúcleos; IV está errada pois sistemas monolíticos não tem uma manutenção fácil e sim complexa.