

1. O que diferencia o núcleo do restante do sistema operacional?  
Resposta: ele gerencia os recursos, ele tem acesso direto aos recursos de hardware, gestão de processos e tarefas
2. Seria possível construir um sistema operacional seguro usando um processador que não tenha níveis de privilégio? Por quê?  
Resposta: não, porque não haveria o isolamento de processos, controle de acessos e proteção do kernel
3. Os processadores da família x86 possuem dois bits para definir o nível de privilégio, resultando em 4 níveis distintos. A maioria dos sistemas operacionais para esses processadores usam somente os níveis extremos (0 e 3, ou 002 e 112 ). Haveria alguma utilidade para os níveis intermediários?  
Resposta: eles são utilizados para a virtualização em sistemas operacionais
4. Quais as diferenças entre interrupções, exceções e traps?  
Resposta: interrupções: são eventos externos ou eventos gerados por periféricos ou dispositivos conectados ao sistema  
Exceções: são eventos gerados pelo próprio processador em resposta a condições anormais ou erros detectados durante a execução de instruções  
Traps: são instruções intencionalmente inseridas no código do programa ou do sistema operacional para gerar um evento específico
5. O comando em linguagem C fopen é uma chamada de sistema ou uma função de biblioteca? Por quê?  
Resposta: é uma função de biblioteca
6. 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? (a) Ler uma porta de entrada/saída (b) Efetuar uma divisão inteira (c) Escrever um valor em uma posição de memória (d) Ajustar o valor do relógio do hardware (e) Ler o valor dos registradores do processador (f) Mascaram uma ou mais interrupções  
Resposta: A) não deveria ser permitida, pois a leitura de portas de entrada/saída normalmente requer acesso direto a hardware específico  
C) Escrever diretamente na memória pode levar a corrupção de dados, erros de execução e instabilidade do sistema  
D) Ajustar o relógio do hardware pode afetar a sincronização do Sistema  
E) A leitura direta dos registradores do processador pode fornecer informações sensíveis sobre o estado do sistema e potencialmente ser explorada para obter informações sobre a arquitetura do processador.  
F) O mascaramento de interrupções é uma ação que pode interferir com o funcionamento correto do sistema operacional e causar problemas de escalonamento
7. 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: (a) Ler o relógio de tempo real do hardware. (b) Enviar um pacote

através da rede. (c) Calcular uma exponenciação. (d) Preencher uma área de memória do processo com zeros. (e) Remover um arquivo do disco.

A) Essa ação provavelmente exigiria uma chamada de sistema. A leitura do relógio de tempo real do hardware envolve acesso direto ao hardware do Sistema

B) Essa ação exigiria uma chamada de sistema. Enviar pacotes através da rede envolve operações de E/S de rede que normalmente são controladas pelo sistema operacional

C) Geralmente, essa ação não requer uma chamada de sistema. Calcular uma exponenciação é uma operação matemática comum que pode ser realizada pelo próprio processo

D) Essa ação geralmente não requer uma chamada de sistema. Preencher uma área de memória com zeros é uma operação de manipulação de memória que pode ser realizada pelo processo sem a necessidade de acesso ao núcleo

E) Essa ação exigiria uma chamada de sistema. Remover um arquivo do disco envolve operações de gerenciamento de arquivos e sistemas de arquivos, que são controladas pelo sistema operacional

8. 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). (a) A rotina de tratamento da interrupção de software é ativada dentro do núcleo. (b) A função printf finaliza sua execução e devolve o controle ao código do processo. Sistemas Operacionais: Conceitos e Mecanismos cap. 2 – pg. 26 (c) A função de biblioteca printf recebe e processa os parâmetros de entrada (a string "Hello world"). (d) A função de biblioteca printf prepara os registradores para solicitar a chamada de sistema write() (e) O disco rígido gera uma interrupção indicando a conclusão da operação. (f) O escalonador escolhe o processo mais prioritário para execução. (g) Uma interrupção de software é acionada. (h) O processo chama a função printf da biblioteca C. (i) A operação de escrita no terminal é efetuada ou agendada pela rotina de tratamento da interrupção. (j) O controle volta para a função printf em modo usuário

H – C- D- G – A- I – E – B – J – F

9. O utilitário strace do UNIX permite observar a sequência de chamadas de sistema efetuadas por uma aplicação. Em um terminal, execute strace date para descobrir quais os arquivos abertos pela execução do utilitário date (que indica a data e hora correntes). Por que o utilitário date precisa fazer chamadas de sistema?

Resposta: As chamadas de sistema são essenciais para permitir que o utilitário **date** obtenha as informações necessárias do sistema operacional

10) O utilitário ltrace do UNIX permite observar a sequência de chamadas de biblioteca efetuadas por uma aplicação. Em um terminal, execute ltrace date para descobrir as funções de biblioteca chamadas pela execução do utilitário date (que indica a data e hora correntes). Pode ser observada alguma relação entre as chamadas de biblioteca e as chamadas de sistema observadas no item anterior?

Resposta: ocorre porque muitas operações de nível superior realizadas pelas funções de biblioteca dependem diretamente das chamadas de sistema subjacentes