

BCC34G – Sistemas Operacionais
Lista de Exercícios #01

1. Quais são as duas principais funções de um sistema operacional ? ^[1]
2. Quais das instruções a seguir deve ser privilegiada? ^[2]
 - a) Configurar o valor do *timer*.
 - b) Ler o relógio.
 - c) Limpar a memória.
 - d) Emitir uma instrução de exceção.
 - e) Desativar interrupções.
 - f) Modificar entradas na tabela de estados de dispositivos.
 - g) Passar da modalidade de usuário para a de núcleo.
 - h) Acessar dispositivo de I/O.
3. O que caracteriza um sistema operacional de tempo real? Quais as duas classificações de sistemas operacionais de tempo real e suas diferenças? ^[3]
4. Qual é a diferença entre modo núcleo e modo usuário? Explique como ter dois modos distintos ajuda no projeto de um sistema operacional. ^[1]
5. Qual tipo de multiplexação (tempo, espaço ou ambos) pode ser usado para compartilhar os seguintes recursos: CPU, memória, disco, placa de rede, impressora, teclado e monitor? ^[1]
6. Para um programador, uma chamada de sistema parece com qualquer outra chamada para uma rotina de biblioteca. É importante que um programador saiba quais rotinas de biblioteca resultam em chamadas de sistema? Em quais circunstâncias e por quê? ^[1]
7. Quais as diferenças entre interrupções de hardware, exceções e interrupções de software (*traps*)? ^[3]
8. Identifique vantagens e desvantagens dos sistemas operacionais de código-fonte aberto. ^[2]
9. Fale sobre como um SO pode ser estruturado: *kernel* (núcleo), *drivers*, código de inicialização, programas utilitários, programas dos usuários.
10. Quais as implicações de mascarar (inibir) interrupções? O que pode ocorrer se o processador ignorar interrupções por muito tempo? O que poderia ser feito para evitar o mascaramento de interrupções? ^[3]
11. O comando em linguagem C *fopen* é uma chamada de sistema ou uma função de biblioteca? Por quê?
12. Monte uma tabela com os benefícios e deficiências mais significativos das principais arquiteturas de sistemas operacionais.
13. O hardware consegue distinguir se as instruções são do SO ou se é de uma aplicação qualquer?
14. Como as instruções de um programa são colocadas na memória para serem executadas pelo hardware?
15. Um programa compilado para um tipo de arquitetura pode ser executado em outro tipo de arquitetura?
16. Explique o processo da inicialização da máquina até o carregamento do SO.
17. Relacione as afirmações aos respectivos tipos de sistemas operacionais: distribuído (D), multiusuário (M), desktop (K), servidor (S), embarcado (E) ou de tempo real (T): ^[3]
 - ☐ Deve ter um comportamento temporal previsível, com prazos de resposta claramente definidos.
 - ☐ Sistema operacional usado por uma empresa para executar seu banco de dados corporativo.
 - ☐ São tipicamente usados em telefones celulares e sistemas eletrônicos dedicados.
 - ☐ Neste tipo de sistema, a localização física dos recursos do sistema computacional é transparente para os usuários.
 - ☐ Todos os recursos do sistema têm proprietários e existem regras controlando o acesso aos mesmos pelos usuários.
 - ☐ A gerência de energia é muito importante neste tipo de sistema.
 - ☐ Sistema que prioriza a gerência da interface gráfica e a interação com o usuário.

BCC34G – Sistemas Operacionais
Lista de Exercícios #01

- ☐ Construído para gerenciar de forma eficiente grandes volumes de recursos.
- ☐ O MacOS X é um exemplo típico deste tipo de sistema.
- ☐ São sistemas operacionais compactos, construídos para executar aplicações específicas sobre plataformas com poucos recursos.
18. 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). ^[3]
- ☐ A rotina de tratamento da interrupção de software é ativada dentro do núcleo.
- ☐ A função `printf` finaliza sua execução e devolve o controle ao código do processo.
- ☐ A função de biblioteca `printf` recebe e processa os parâmetros de entrada (a string "Hello world").
- ☐ A função de biblioteca `printf` prepara os registradores para solicitar a chamada de sistema `write()`
- ☐ O disco rígido gera uma interrupção indicando a conclusão da operação.
- ☐ O escalonador escolhe o processo mais prioritário para execução.
- ☐ Uma interrupção de software é acionada.
- ☐ O processo chama a função `printf` da biblioteca C.
- ☐ A operação de escrita no terminal é efetuada ou agendada pela rotina de tratamento da interrupção.
- ☐ O controle volta para a função `printf` em modo usuário.
19. Fale sobre os tipos de Sistemas Operacionais e as características principais de cada tipo.
20. Compare SO monoprogramado e multiprogramado segundo a E/S.
21. O que é *buffering* e qual o seu relacionamento com as velocidades da CPU e dos dispositivos de E/S?
22. Explique a técnica de *spooling*?
23. Como o *caching* melhora o desempenho do sistema? ^[4]
24. Como *drivers* e interfaces do tipo *plug-and-play* facilitam a extensibilidade de sistemas operacionais? ^[4]
25. Como um temporizador de intervalo evita que um processo monopolize o processador? ^[4]
26. Como a *Extensive Firmware Interface (EFI)* aborda as limitações da BIOS antigas? ^[4]
27. Como o POSIX tenta melhorar a portabilidade de aplicação? ^[4]
28. O utilitário *strace* do UNIX permite observar a sequência de chamadas de sistema efetuadas por uma aplicação. Em um terminal UNIX, 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? ^[3]
29. O utilitário *ltrace* do UNIX permite observar a sequência de chamadas de biblioteca efetuadas por uma aplicação. Em um terminal UNIX, 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? ^[3]

Referências:

- [1] TANENBAUM, A. S.; BOS, H.. **Sistemas Operacionais Modernos**. 4a ed. Pearson, 2016. págs 65-58.
- [2] SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. **Fundamentos de Sistemas Operacionais**. 9. ed. LTC, 2015. págs. 55-57.
- [3] MAZIERO, C. **Sistemas Operacionais: Conceitos e Mecanismos**. Online. Caderno de Exercícios. 2013. págs 3-6.
- [4] DEITEL, H.; DEITEL, P. J.; CHOFFNES, D. R. **Sistemas Operacionais**. 3. ed. São Paulo: Pearson, 2005.