

## Lista de Exercícios (Primeira parte)

- 1) O que é um Sistema Operacional (SO)?
- 2) Quais são as funções básicas de um SO?
- 3) O que acontece com o conteúdo da memória Ram quando o computador é desligado?
  - a) Permanece armazenado
  - b) É parcialmente apagado
  - c) É totalmente perdido
  - d) É gravado
- 4) Cite 3 dispositivos de entrada e 3 dispositivos de saída.
- 5) Defina Sistema Computacional.
- 6) Quais as características dos SOs monoprogramáveis?
- 7) O que é uma linguagem de programação?
- 8) Quais as desvantagens encontradas em uma linguagem de máquina?
- 9) O que é multiprogramação? Cite duas razões para se ter multiprogramação?
- 10) Há várias metas de projeto na construção de um sistema operacional; por exemplo, utilização de recursos, oportunidade, robustez etc. Dê um exemplo de duas metas de projeto que possam ser contraditórias.
- 11) Qual a finalidade de uma chamada de sistema em um sistema operacional?
- 12) Considere um sistema de computador que tem memória cache, memória principal (RAM) e disco. O sistema operacional usa memória virtual. São necessários 2ns para acessar uma palavra a partir da cache, 10ns para acessar uma palavra a partir da RAM e 10ms para acessar uma palavra a partir do disco. Se a taxa de acerto da cache é de 95% e a da memória principal (após uma falta de cache) é de 99%, qual é o tempo médio de acesso a uma palavra?
- 13) Imagine que um arquivo de 10MB esteja armazenado em um disco na mesma trilha (trilha #:50) em setores consecutivos. O braço do disco está situado sobre a trilha de número 100. Quanto tempo é necessário para recuperar esse arquivo a partir do disco? Suponha que a transferência do braço de um cilindro a outro leve cerca de 1ms e cerca de 5ms para que o setor onde o início do arquivo está armazenado faça a rotação sob a cabeça. Além disso, suponha que a leitura ocorra a uma taxa de 100MB/s.
- 14) Como a computação interativa e a melhoria que causou no tempo de retorno aumentaram a produtividade do programador?
- 15) Como a tecnologia orientada a objeto afetou os sistemas operacionais?
- 16) Cite alguns dos benefícios do desenvolvimento de software livre.
- 17) Quais tecnologias podem ser usadas para transpor a lacuna entre sistemas operacionais diferentes? Como essas tecnologias possibilitaram a execução da mesma aplicação em várias plataformas?

- 18) Um monitor de temperatura em uma usina nuclear seria provavelmente descrito como que tipo de sistema? Por quê?
- 19) Descreva as vantagens e desvantagens das máquinas virtuais.
- 20) Em um ambiente de multiprogramação e tempo compartilhado, vários usuários compartilham o sistema simultaneamente. Essa situação pode resultar em diversos problemas de segurança.
- Cite dois desses problemas.
  - Podemos assegurar o mesmo nível de segurança tanto em uma máquina dedicada como em uma máquina de tempo compartilhado? Explique sua resposta.
- 21) Por que a separação entre mecanismo e política é desejável?
- 22) Qual é a finalidade dos programas de sistema?
- 23) Qual a maior vantagem de implementar threads no espaço do usuário? Qual a maior desvantagem?
- 24) Descreva as diferenças entre o scheduling de curto prazo, de médio prazo e de longo prazo.

- 25) Incluindo o processo pai inicial, quantos processos são criados pelo programa mostrado abaixo?

```
#include<stdio.h>
#include<unistd.h>
int main()
{
    fork();
    fork();
    fork();
    return 0;
}
```

- 26) A sequência de Fibonacci é a série de números 0, 1, 1, 2, 3, 5, 8, ... Formalmente, ela pode ser expressa como:

$$\begin{aligned} fib_0 &= 0 \\ fib_1 &= 1 \\ fib_n &= fib_{n-1} + fib_{n-2} \end{aligned}$$

Escreva um programa usando a chamada de sistema `fork()` que gere a sequência de Fibonacci no processo filho. O número da sequência será fornecido na linha de comando. Por exemplo, se 5 for fornecido, os cinco primeiros números da sequência serão exibidos pelo processo filho. Já que os processos pai e filho têm suas próprias cópias dos dados, será necessário que o filho exiba a sequência. Faça o pai invocar a chamada `wait()` para esperar o processo filho ser concluído antes de sair do programa. Execute a verificação de erros necessária para assegurar que um número negativo seja passado na linha de comando.

- 27) Forneça dois exemplos de programação em que a criação de vários threads proporcione melhor desempenho do que uma solução com um único thread.
- 28) Cite duas diferenças entre os threads de nível de usuário e os de nível de kernel. Sob que circunstâncias um tipo é melhor do que o outro?
- 29) Projete um programa de cópia de arquivos chamado `FileCopy` usando pipes comuns. Esse programa receberá dois parâmetros: o primeiro é o nome do arquivo a ser copiado e o segundo é o nome do arquivo copiado. Em seguida, o programa criará um pipe comum e gravará nele o conteúdo do arquivo a ser copiado. O processo filho lerá esse arquivo do pipe e o gravará no arquivo destino. Por exemplo, se chamarmos o programa como descrito a seguir: `FileCopy input.txt copy.txt`  
O arquivo `input.txt` será gravado no pipe. O processo filho terá o conteúdo desse arquivo e o gravará no arquivo de destino `copy.txt`. Você pode escrever esse programa usando pipes Windows ou Unix.

- 30) Implemente uma comunicação Cliente-Servidor utilizando sockets. Nesta comunicação o servidor deverá transferir para o cliente um arquivo txt contendo informações (livre escolha, por exemplo: valores de data, hora e minuto) atualizadas.
- 31) Modifique o servidor baseado em socket (exercício 8) para que o serviço atenda cada solicitação de cliente em um thread separado.
- 32) Modifique o exercício sobre a sequência de Fibonacci (exercício 4) para um programa com vários threads que gere a sequência, a partir de um que o usuário deverá inserir na linha de comando.
- 33) A **solução de Peterson** é uma solução clássica baseada em software para o problema da seção crítica. Descreva o funcionamento deste algoritmo e demonstre este funcionamento com um exemplo.
- 34) Suponha que sejam criados 5 processos. Utilize semáforos para garantir que o processo 1 escreva os números de 1 até 100, o processo 2 escreva os números de 101 até 200, e assim por diante.  
Dica: o processo  $i+1$  só deve entrar em funcionamento quando processo  $i$  já tiver terminado a escrita dos seus números  
Descreva uma solução para este problema. Apresente apenas a sincronização de execução dos processos utilizando pseudocódigo.
- 35) Descreva o problema de sincronização dos **Leitores-Gravadores** e apresente uma possível estrutura, utilizando semáforos, para o um processo leitor e para o um processo gravador.
- 36) Descreva o problema do **jantar dos filósofos** e uma possível solução utilizando semáforos e outra solução utilizando monitores.
- 37) Descreva o problema do **barbeiro adormecido** e faça um programa para coordenar o barbeiro e os clientes.

### **(Segunda parte)**

- 38) Cite duas diferenças entre endereços lógicos e físicos.
- 39) Em um sistema com paginação, um processo não pode acessar memória que ele não possui. Por quê? Como o sistema operacional poderia permitir o acesso a outra memória? Por que ele deveria ou não deveria fazer isto?
- 40) Por que a segmentação e a paginação são algumas vezes combinadas em um esquema?
- 41) Sob que circunstâncias ocorrem erros de páginas (falta de página)? Descreva as ações executadas pelo sistema operacional quando ocorre um erro de página.
- 42) Considere a seguinte sequência de referências de páginas:  
*1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6*  
Quantos erros de página ocorreriam para os algoritmos de substituição a seguir, considerando um, dois, três, quatro, cinco, seis e sete quadros? Lembre-se de que todos os quadros estão inicialmente vazios, de modo que a primeira página de cada um implicará em um erro de página.
  - a) Substituição LRU
  - b) Substituição FIFO
- 43) Qual é a causa da atividade improdutivo? Como o sistema detecta a atividade improdutivo? Uma vez que ela seja detectada, o que o sistema pode fazer para eliminar este problema?

- 44) Escreva um programa que implemente os algoritmos de substituição de páginas FIFO e LRU. Primeiro, gere uma sequência de referências de páginas aleatória em que os números de página variam de 0 a 9. Aplique a sequência de referências de página ocorridos em cada algoritmo. Implemente os algoritmos de substituição de modo que o número de quadros de páginas possa variar de 1 a 7. Considere o uso da paginação por demanda.
- 45) Por que alguns sistemas administram o tipo do arquivo enquanto outros delegam essa função ao usuário e outros simplesmente não implementam múltiplos tipos de arquivos? Qual sistema é “melhor”?
- 46) De modo semelhante, alguns sistemas suportam muitos tipos de estruturas para os dados de um arquivo enquanto outros suportam simplesmente uma cadeia de bytes. Quais são as vantagens e desvantagens de cada abordagem?
- 47) Dê um exemplo de uma aplicação em que os dados de um arquivo devam ser acessados na seguinte ordem:
- a) Seqüencialmente
  - b) Aleatoriamente
- 48) Dê um exemplo de aplicação que pudesse se beneficiar do suporte do sistema operacional ao acesso aleatório a arquivos indexados.
- 49) Um disco CD-ROM contém um sistema de arquivos no qual todos os arquivos são imutáveis. Qual método de alocação, entre alocação contígua, encadeada e indexada, é o mais apropriado? Justifique sua resposta.
- 50) O que acontece quando se apaga a MBR (Master Boot Record) de um disco principal?
- 51) O que são listas de controle de acessos?
- 52) Pesquise e descreva sobre os sistemas de arquivos do Linux, Mac OS e Windows.

Boa Sorte! ☺