

Sistemas Operacionais

1. O que é um sistema operacional?
2. Quais são as principais funções de um sistema operacional?
3. Quais são os tipos de sistemas operacionais existentes?
4. Por que em sistemas monotarefa (ou monoprogramáveis), dizemos que há subutilização dos recursos?
5. Qual é a grande diferença entre os sistemas monotarefa e os multitarefa?
6. Um sistema multiusuário pode ser monotarefa?
7. Quais são as principais características de um sistema batch?
8. Como funcionam os sistemas de tempo compartilhado?
9. Quais são as vantagens em usar sistemas de tempo compartilhado?
10. O que é fatia de tempo?
11. Como funciona a técnica de reentrância?
12. Qual a definição de um sistema preemptível e não-preemptível?
13. Qual é a diferença entre um processo CPU-Bound e I/O Bound?
14. Como podemos descrever a atividade de Escalonamento de Processos?
15. Por que precisamos utilizar a Comunicação entre Processos? Cite uma forma de comunicação.
16. Sobre os algoritmos de escalonamento, quais seus princípios fundamentais e como funciona o algoritmo de escalonamento Round Robin?
17. Como se dá o escalonamento lotérico?
18. Como podemos definir o *kernel*?
19. Qual as características que distinguem um processo de uma thread?
20. Como podemos realizar a exclusão mútua?
21. O que é uma região crítica?
22. Defina deadlock e starvation.
23. Defina os estados que um processo pode assumir e a relação entre esses estados e as interrupções.
24. Qual a definição de Chamada de Sistema?
25. Cite exemplos de chamadas de sistema utilizados no sistema operacional Linux.

1. O que é um sistema operacional?

Ele é o primeiro software a ser carregado pelo computador e desempenha várias funções importantes. É o software básico do computador, sendo responsável por gerenciar o hardware, os programas e a interação entre os programas e o hardware.

Ele, ainda, organiza o funcionamento do computador gerenciando os recursos do sistema como a memória RAM e o processador, dividindo e controlando a utilização desses recursos pelos programas em execução.

2. Quais são as principais funções de um sistema operacional?

- inicializar o hardware do computador
- fornecer rotinas básicas para controle de dispositivos
- fornecer gerência, escalonamento e interação de tarefas
- manter a integridade de sistema

As duas principais funções são “facilidade de acesso aos recursos do sistema” e “compartilhamento de recursos de forma organizada e protegida”.

- a) Gerenciar compartilhamento e acesso seguro o hardware (recurso) disponível.
- b) Disponibilizar uma interface (API) para os processos.
- c) Executar e fornecer recursos as processos.

3. Quais são os tipos de sistemas operacionais existentes?

SO de Computadores Pessoais;
SO de Computadores de Grande Porte;
SO de Servidores;
SO de Tempo Real;
SO de Embarcados;
SO de Cartões Inteligentes;

4. Por que em sistemas monotarefa (ou monoprogramáveis), dizemos que há subutilização dos recursos?

Porque quando qualquer programa executado, ele deveria aguardar o termino do programa corrente. Ele permitia que o processador, a memória e os periféricos permaneçam dedicados apenas para a execução de um único programa. Quando um programa aguarda, por exemplo, a digitação de um dado, o processador permanece ocioso, sem realizar qualquer tipo de processamento. É por isso que a memória acabava sendo subutilizada.

5. Qual é a grande diferença entre os sistemas monotarefa e os multitarefa?

A diferença é a capacidade de execução de diversos programas ao mesmo tempo.

O sistema que permite a execução de vários programas simultaneamente é conhecido como "Multitarefas". Nele os programas com a mesma prioridade de tempo rodam um de cada vez, dando a aparência de que tudo está sendo executado ao mesmo tempo. No sistema operacional monotarefa não suporta mais de um programa rodando ao mesmo.

6. Um sistema multiusuário pode ser monotarefa?

Sim, somente um usuário interage com o sistema podendo possuir diversas aplicações executando concorrentemente.

7. Quais são as principais características de um sistema batch?

Pode ser executados cálculos numéricos, backups, compilações entre outros. Um bom exemplo de operação que é executada em Batch é a compensação bancária.

O processamento batch tem a característica de não exigir a interação do usuário com a aplicação. Todas as entradas e saídas de dados da aplicação são implementadas por algum tipo de memória secundária, geralmente arquivos em disco. Em outras palavras, caracteriza-se por um processamento de dados que ocorre através de um lote de tarefas enfileiradas, de modo que o sistema operacional só processa a próxima tarefa após o término completo da tarefa anterior.

Ele se caracteriza por ter programas executados de forma sequencial. Nele, os programas não exigem interação do usuário, lendo e gravando dados em discos.

8. Como funcionam os sistemas de tempo compartilhado?

Os sistemas de tempo compartilhado (timesharing) dividem o poder computacional do computador entre os usuários ao mesmo tempo, utilizando-se de terminais de vídeo e possuem arquitetura complexa.

Nele o tempo do processador é dividido em fatias, com isso uma tarefa pode ser executada em mais de uma dessas fatias, não consecutivas. Ele também permite interação com o usuário.

9. Quais são as vantagens em usar sistemas de tempo compartilhado?

Ele permite a interação dos usuários com o sistema, através de terminais que incluem vídeo, teclado e mouse. Eles possuem uma linguagem de controle que permite ao usuário comunicar-se diretamente com o sistema operacional, através de comandos.

Desta forma, é possível verificar arquivos armazenados em disco ou cancelar a execução de um programa. O sistema, normalmente, responde em poucos segundos à maioria desses comandos. Devido a esse tipo de interação, os sistemas de tempo compartilhado também ficaram conhecidos como sistemas on-line.

10. O que é fatia de tempo?

Fatia de tempo do processo. A fatia de tempo é o valor numérico que representa por quanto tempo um processo pode ser executado até que seja suspenso. A fatia de tempo, de fato, fornece a cada processo uma fatia do tempo de uso do processador.

11. Como funciona a técnica de reentrância?

Quando vários usuários utilizarem os mesmos aplicativos simultaneamente, se cada usuário que utilizasse um destes aplicativos trouxesse o código executável para a memória, haveria então diversas cópias de um mesmo programa ocupando espaço na memória, o que causaria um grande desperdício de espaço. Para que isso não aconteça, o código executável é compartilhado por vários usuários, exigindo apenas uma cópia do programa em memória. A reentrância permite que cada usuário esteja executando um trecho diferente do código reentrante, manipulando dados próprios.

12. Qual a definição de um sistema preemptível e não-preemptível?

Não preemptivo: O processo executa até o fim, sem ser interrompido;

Preemptivo: O processo executa em fatias de tempo (quantum) determinado pelo sistema operacional.

13. Qual é a diferença entre um processo CPU-Bound e I/O Bound?

CPU-Bound: É um processo que utiliza muita CPU. O seu tempo de execução é definido principalmente pelo tempo dos ciclos(clocks) do processador. Esse tipo de processo realiza poucas operações de leitura e gravação e é encontrado em aplicações científicas e de muito cálculo. Por exemplo um processo que executa um programa de inversão de matriz é cpu-bound. Após ler alguns poucos dados, ele precisa apenas de processador.

I/O-Bound: Um processo é considerado I/O-Bound quando passa a maior parte do tempo no estado de espera, pois realiza um elevado número de operações de E/S. Nesse caso, o tempo de execução é definido principalmente por estas operações. Por exemplo, um processo que executa um programa de cópia de arquivo é I/O-Bound. Ele praticamente não utiliza processador, apenas acessa disco. Este tipo de processo é encontrado principalmente em aplicações comerciais, que se baseiam em leitura, processamento e gravação.

14. Como podemos descrever a atividade de Escalonamento de Processos?

A atividade de escalonamento de processos em sistemas multiprocessadores e multicomputadores (adotados em arquiteturas paralelas e sistemas computacionais distribuídos) consiste em atribuir processos a elementos de processamento (processadores). Esse tipo de escalonamento é conhecido como escalonamento global contraposição ao escalonamento local realizado pelo sistema operacional em uma única máquina.

15. Por que precisamos utilizar a Comunicação entre Processos? Cite uma forma de comunicação.

Uma forma de evitar que processos concorrentes acessem o mesmo endereço de memória ao mesmo tempo. O Compartilhamento de Memória é um mecanismo de comunicação entre processos que usa uma área de memória, um buffer que é compartilhado entre os vários processos de uma aplicação concorrente. Os processos compartilham informações em operações de escrita e leitura através do buffer de memória.

16. Sobre os algoritmos de escalonamento, quais seus princípios fundamentais e como funciona o algoritmo de escalonamento Round Robin?

Um dos mais antigos, simples, justos, e mais largamente utilizados algoritmo de escalonamento é o Round Robin. Cada processo recebe um intervalo de tempo, chamado quantum, durante o qual ele pode executar. Se o processo ainda estiver executando ao final do quantum, o sistema operacional interrompe a sua execução e passa a UCP a outro processo. Se um processo bloqueou ou terminou antes do final do quantum, a troca da UCP para outro processo é obviamente feita assim que o processo bloqueia ou termina. Round Robin é fácil de implementar. Tudo que o escalonador tem a fazer é manter uma lista de processos que desejam executar.

17. Como se dá o escalonamento lotérico?

A ideia básica é dar bilhetes de loteria aos processos para vários recursos do sistema, como o tempo da CPU. Sempre que uma decisão de escalonamento tiver de ser feita, um bilhete de loteria será escolhido ao acaso, e o processo com o bilhete fica com o recurso. Quando aplicado ao escalonamento de CPU, o sistema pode realizar um sorteio 50 vezes por segundo, com cada vencedor recebendo 20 ms de tempo da CPU como prêmio.

O escalonamento de loteria tem várias propriedades interessantes. Por exemplo, se um novo processo aparece e ele ganha alguns bilhetes, no sorteio seguinte ele teria uma chance de vencer na proporção do número de bilhetes que tem em mãos. Em outras palavras, o escalonamento de loteria é altamente responsivo.

18. Como podemos definir o kernel?

É um conjunto de rotinas (Procedimentos) que oferecem serviços aos usuários do sistema e suas aplicações.

19. Qual as características que distinguem um processo de uma thread?

Um thread (ou processo leve) é uma unidade básica de execução na CPU, que consiste em apontador de instruções, conjunto dos registradores e espaço de pilha, compartilhando entre si: área de código, área de dados, recursos do sistema operacional (tarefa). Um processo é equivalente a uma tarefa, já a thread é uma única tarefa do processo.

20. Como podemos realizar a exclusão mútua?

Existem várias propostas para realizar a exclusão mútua, de maneira que enquanto um processo está ocupado atualizando a memória compartilhada em sua região crítica, nenhum outro entrará na sua região crítica para causar problemas. Para obtenção de exclusão mútua pode-se utilizar: desabilitando interrupções, variáveis do tipo trava, chaveamento obrigatório, solução de Peterson, a instrução TSL, entre outros.

21. O que é uma região crítica?

Seção do programa onde são efetuados acessos (para leitura e escrita) a recursos compartilhados por dois ou mais processos. É necessário assegurar que dois ou mais processos não se encontrem simultaneamente na região crítica

22. Defina deadlock e starvation.

Deadlock: um problema que está presente em todos os sistemas operacionais atuais, que leva ao travamento de processos.

Starvation: a situação de starvation está presente quando o sistema operacional provê prioridades a processos, que não atualizados fazem com que os processos de menor prioridade nunca sejam executados causando assim, deficiência em servidores de impressão e etc

23. Defina os estados que um processo pode assumir e a relação entre esses estados e as interrupções.

É o mecanismo programático pelo qual um programa de computador solicita um serviço do núcleo do sistema operacional sobre o qual ele está sendo executado. Isto pode incluir serviços relacionados ao hardware (por exemplo, acessar uma unidade de disco rígido), criação e execução de novos processos e comunicação com serviços do núcleo de maneira integral como escalonamento do processador. Chamadas do sistema fornecem uma interface essencial entre um processo e o sistema operacional.

24. Qual a definição de Chamada de Sistema?

São funções do S.O. acessíveis pelos processos de usuário. Funcionam como chamada a funções comuns de qualquer biblioteca. Porém, são executadas pelo Kernel do S.O. e, normalmente, exigem estar em modo Kernel para tal. As principais funções de uma chamada de sistema: gerência de processos e threads; gerência de sistema de arquivos e gerência de memória.

25. Cite exemplos de chamadas de sistema utilizados no sistema operacional Linux.

- kill
- chmod
- mount
- umount
- link
- execve