



Checkpoint II

Questão 1

Qual dos seguintes não é um objetivo fundamental dos Sistemas Operacionais para este dispositivo inteligente?

Resposta: d) Eliminar completamente a necessidade de drivers de dispositivo.

Questão 2

No contexto dos Sistemas Operacionais, qual a função essencial dos "DRIVERS" em relação à comunicação e gerenciamento do hardware, que permite ao SO interagir com uma vasta gama de periféricos de forma padronizada?

Resposta: b) Atuam como uma camada de abstração, contendo as informações sobre o que o equipamento é capaz de realizar e os códigos necessários para acessá-lo.

Questão 3

Qual a principal desvantagem dos Sistemas Monoprogramáveis ou Monotarefa, como essa ineficiência impactava severamente a utilização dos recursos computacionais?

Resposta: b) Nesses sistemas, o processor permanecia ocioso enquanto aguardava por um evento de Entrada/Saída, resultando em uma baixa utilização da CPU e da memória principal.

Questão 4

Qual a distinção essencial entre Sistemas Fortemente Acoplados e Sistemas Fracamente Acoplados?

Resposta: b) Fortemente acoplados compartilham uma memória comum e são mais eficientes, enquanto fracamente acoplados se comunicam por rede e são mais tolerantes a falhas.

Questão 5

A técnica de Entrada e Saída controlada por programa ou Polling, como descrita, é ineficiente porque?

Resposta: b) Gerando um enorme desperdício de tempo da CPU poder ser usado em outras tarefas.

Questão 6

Como a implementação de mecanismo de interrupção nos operacionais permite uma comparação com a técnica de polling, permitindo a multiprogramação e a reatividade em sistemas modernos?

Resposta: b) Permitindo a substituição do polling por interrupções, onde a CPU é notificada quando uma operação de E/S termina, liberando-a para outras tarefas durante a espera.

Questão 7

Um desenvolvedor de software está depurando um programa complexo que, ocasionalmente, exibe comportamentos inesperados, como crashes ou falhas de segurança. Para entender o que acontece, ele precisa monitorar o fluxo de execução do programa e o sistema operacional. Qual a diferença central entre uma Interrupção (de Hardware) e uma Exceção (de Software), em termos de sua origem e como o sistema operacional responde a elas?

Resposta: b) A interrupção é um evento imprevisível, mas a exceção é o resultado direto de uma instrução específica no código do programa (como uma divisão por zero).

Questão 8

Qual técnica de Entrada e Saída permite que o E/S seja feito de forma independente da CPU, liberando a CPU para outras tarefas durante a maior parte da operação?

Resposta: c) Acesso Direto a Memória (DMA), onde um controlador dedicado gerencia a transferência de blocos de dados.

Questão 9

Qual a diferença fundamental entre uma Interrupção, uma Exceção e uma Trap no contexto de sistemas operacionais?

Resposta: a) Uma Interrupção é assíncrona, gerada por hardware externo; uma Exceção é síncrona, gerada por erro no programa; uma Trap é síncrona, intencional para chamar o SO.

Questão 10

Qual o mecanismo principal que impede que um programa de usuário execute operações privilegiadas diretamente, forçando-o a solicitar ao sistema operacional que o faça em seu nome?

Resposta: a) Modo de Usuário (User Mode), onde o programa é executado com privilégios limitados, e o Modo de Supervisor (Kernel Mode), onde o SO executa operações privilegiadas.

Questão 11

Qual mecanismo presente nos processadores permite que um software realize uma operação que exija acesso privilegiado, como acessar um dispositivo de hardware ou modificar configurações do sistema?

Resposta: c) A reescrita do programa de aplicação em linguagem de assembly para incluir instruções privilegiadas.

Questão 12

Uma startup está projetando um novo sistema operacional para dispositivos IoT muito simples e de baixo custo, onde a performance e a simplicidade são prioritárias. Eles precisam decidir qual arquitetura de kernel é a mais adequada para esse cenário, considerando que o sistema será executado em hardware com recursos limitados.

Resposta: b) Arquitetura Microkernel, que separa os serviços em processos independentes.

Questão 13

Uma empresa de desenvolvimento de sistemas busca criar um sistema operacional de alta disponibilidade e robustez, onde a falha de um serviço não comprometa o todo. Eles consideram diferentes arquiteturas, mas priorizam a modularidade e a facilidade de manutenção.

Resposta: b) Arquitetura em Camadas, que organiza os serviços em níveis hierárquicos.

Questão 14

Em um sistema operacional multitarefa, a ilusão de que vários programas estão sendo executados simultaneamente é criada através do gerenciamento eficiente dos recursos do sistema, particularmente da CPU. Para que o SO possa alternar entre diferentes tarefas de forma eficiente, ele precisa manter informações sobre cada tarefa em andamento.

Resposta: c) Um processo é apenas o código executável de um programa, e seus componentes são input, Output e Memory.

Questão 15

Em um ambiente computacional moderno e multitarefa, o sistema operacional gerencia a execução de múltiplos programas de forma que pareça que eles estão rodando simultaneamente. Para conseguir isso, ele precisa de uma ilusão de continuidade para cada programa em execução, mesmo quando ele é interrompido para dar vez a outro.

Resposta: b) Bloco de Controle de Processo (PCB), que contém informações como o estado do processo, contadores de programa, alocação de memória e mais.

Questão 16

Em um sistema multiprogramável, um programa em execução em executed pode assumir diferentes estados primários e mais comum sistema operacional multiprogramável, considerando as transições voluntárias e involuntárias?

Resposta: a) Iniciado, Em Pausa, Terminado.

Questão 17

Em um sistema operacional de tempo compartilhado, múltiplos processos competem pelo uso da CPU. Para garantir que nenhum processo monopolize o processador, o sistema implementa um mecanismo de preempção baseado em tempo.

Resposta: b) O uso de um timer para gerar interrupções em intervalos regulares, permitindo que o SO alterne entre processos.

Questão 18

Um grupo de estudantes de engenharia de software está desenvolvendo um aplicativo complexo que envolve a criação de vários módulos que devem funcionar de forma cooperativa, mas com diferentes níveis de isolamento e compartilhamento de recursos. Eles debatem se devem implementar esses módulos como processos independentes ou como subprocessos.

Resposta: c) Subprocessos são sempre mais rápidos que processos independentes.

Questão 19

Uma empresa de desenvolvimento de jogos esta otimizando um motor gráfico para rodar em hardware limitado, como consoles ou dispositivos móveis. Para

renderizar frames e processar a lógica do jogo simultaneamente, eles precisam de um mecanismo que permita a execução paralela de tarefas dentro do mesmo programa, aproveitando o máximo os múltiplos núcleos do processador.

Resposta: a) Threads são mais seguros, pois cada um possui seu próprio espaço de endereçamento isolado.

Questão 20

Uma equipe de desenvolvedores está criando um novo sistema operacional e debate a forma de implementar o suporte a threads. Eles precisam decidir se o suporte será fornecido pelo kernel do SO ou pela biblioteca de usuário. A escolha impacta na portabilidade, eficiência e complexidade do sistema.

Resposta: c) Threads em Modo Usuário (TMU) e "Threads em Modo Kernel" (TMK), onde o TMU é implementado pela biblioteca do usuário e o TMK pelo kernel do SO.

Questão 21

Um sistema de gerenciamento de tráfego aéreo existe dois tipos de módulos: um que realiza cálculos complexos de trajetória de voo, exigindo muita capacidade de processamento da CPU, e outro que monitora constantemente os radares e sensores, exigindo respostas rápidas a eventos externos. Para otimizar o sistema, eles precisam categorizar corretamente esses módulos.

Resposta: a) Arquitetura em Camadas, que isola as funções em níveis hierárquicos.

Questão 22

Em um sistema operacional multitarefa, é comum ter vários programas prontos para atender-los, mas a CPU pode executar apenas um de cada vez. Para garantir que todos os programas sejam atendidos de forma eficiente e que o sistema permaneça responsivo, o SO utiliza uma estratégia para decidir qual programa executar em seguida.

Resposta: b) O escalonador seleciona o processo com o menor tempo restante de execução.

Questão 23

Uma empresa de processamento de dados opera um servidor que executa tarefas de análise financeiras em lote. A gerência deseja maximizar a

produtividade do sistema em termos de conclusão de tarefas, priorizando as tarefas que podem ser completadas mais rapidamente para liberar recursos.

Resposta: a) O escalonador prioriza tarefas com o menor tempo de execução restante.

Questão 24

Qual critério de escalonamento mede a quantidade de processos que são efetivamente concluídos em um dado intervalo de tempo, sendo uma métrica útil para avaliar a eficiência do sistema em um ambiente de processamento em lote?

Resposta: b) Tempo de Processador (CPU Burst), que é o tempo total que um processo leva para ser executado.

Questão 25

Qual critério de escalonamento representa o tempo total desde a submissão de um processo até sua conclusão, refletindo a percepção do usuário sobre o desempenho do sistema?

Resposta: a) Tempo de Turnaround.

Questão 26

Dois administradores de sistemas debatem a escolha de uma política de escalonamento para um servidor que atende a múltiplos usuários. Um defende uma política que permite a execução completa de um processo antes de passar para o próximo, enquanto o outro prefere uma abordagem que alterna entre processos para garantir que todos tenham oportunidade de progresso.

Resposta: c) O algoritmo de escalonamento FIFO (First In, First Out), também conhecido como FCFS (First Come, First Served), segue qual princípio essencial para selecionar o próximo processo a ser executado, e qual é a principal característica em relação ao ordem de chegada?

Questão 27

Um laboratório de processamento de imagens tem uma série de tarefas de diferentes durações. Se a política de escalonamento sempre preferir as tarefas mais curtas, há um risco de que as tarefas mais longas nunca sejam executadas ou sejam atrasadas indefinidamente.

Resposta: c) Starvation (Adiamento Indefinido), a possibilidade de um processo ser adiado indefinidamente, correndo o risco de nunca ser executado.

Questão 28

Um administrador de sistemas está otimizando o desempenho de um servidor que executa múltiplas aplicações simultaneamente. Ele observa que o sistema está alocando memória de forma ineficiente, levando a gargalos e falhas.

Resposta: b) Para ocultar o custo da memória principal, que é um recurso escasso, o sistema operacional deve priorizar a memória para maximizar o compartilhamento do processador e de outros recursos computacionais.

Questão 29

Um sistema operacional legado, a memória principal é dividida em uma área para o sistema operacional e outra para o programa do usuário. Como não há suporte a multiprogramação, apenas um programa pode ser executado de cada vez. Essa simplicidade, entretanto, leva a uma utilização ineficiente dos recursos, especialmente da memória.

Resposta: d) A complexidade na implementação do controle de acesso à memória.

Questão 30

Uma equipe de desenvolvimento está otimizando um sistema operacional embarcado e precisa decidir sobre uma estratégia de alocação de memória que minimize o desperdício e maximize a utilização, mesmo com programas de tamanhos variados. Eles consideram diferentes algoritmos, mas priorizam a simplicidade e a velocidade.

Resposta: b) Deadlock, um bloqueio mútuo entre processos que disputam recursos.

Questão 31

Um cientista de dados está executando um algoritmo complexo que manipula um enorme volume de dados. Ele percebe que seu programa, que exige gigabytes de RAM se carregar, está rodando lentamente, com frequentes pausas. O sistema operacional está inteligente para gerenciar a memória, trocando partes do programa entre a memória e o disco para permitir que outros programas rodem.

Resposta: a) A impossibilidade de proteger a área do sistema operacional de acessos indevidos.

Questão 32

Um sistema que utiliza memória virtual por paginação, um programa tenta acessar uma página que não está na memória principal na RAM naquele momento. Essa situação dispara um mecanismo para trazer a página do disco para a memória, permitindo que o programa continue.

Resposta: b) Um "page fault", que faz com que o sistema transfira a página da memória secundária para a principal.

Questão 33

Um engenheiro de desempenho está ajustando os parâmetros de um sistema operacional para melhorar a responsividade de aplicativos que exigem muita memória. Ele precisa garantir que o sistema não sofra com interrupções excessivas causadas por acessos a páginas que não estão na memória principal.

Resposta: a) Apenas quando o processo termina ou é interrompido.

Questão 34

Um analista de sistemas percebe que a máquina virtual de um servidor está com baixa performance, pois uma parte do código é executada repetidamente, acessando a mesma área de memória. O hardware do processador possui um mecanismo para acelerar esse acesso, reduzindo o tempo de busca na memória.

Resposta: d) Cache de CPU, que armazena dados frequentemente acessados para acelerar o processamento.

Questão 35

Qual termo é utilizado para descrever a situação de excessiva transferência de páginas/segmentos entre a memória principal e a memória secundária, que ocorre quando há mais processos competindo pela memória do que o sistema pode suportar de forma eficiente?

Resposta: d) Thrashing, a troca excessiva de páginas que degrada o desempenho devido à intensa atividade de paginação/swapping.

Questão 36

Uma equipe de desenvolvimento está projetando um sistema que lidará com grandes volumes de dados em memória. Para otimizar o acesso, eles consideram diferentes estruturas de dados, como arrays, listas ligadas ou árvores. A escolha afeta a eficiência em termos de tempo de busca e atualização.

Resposta: e) O uso exclusivo de TLB, que otimiza o acesso à memória.