

Exercícios Cap 3

- 1)
 - Monolítico
 - Benefícios: Desempenho rápido, baixa sobrecarga de comunicação, implementação simples.
 - Deficiências: Falta de modularidade, falta de isolamento de falhas.
 - Microkernel
 - Benefícios: Alta modularidade, isolamento de falhas, escalabilidade.
 - Deficiências: Overhead de comunicação, complexidade na implementação.
 - Cliente-Servidor
 - Benefícios: Separação clara, distribuição de recursos, escalabilidade.
 - Deficiências: Overhead de comunicação de rede, complexidade na coordenação.
 - Máquinas Virtuais
 - Benefícios: Isolamento, migração de sistemas, segurança.
 - Deficiências: Overhead de virtualização, possível impacto no desempenho.
 - Sistemas Embarcados
 - Benefícios: Footprint reduzido, inicialização rápida, otimização.
 - Deficiências: Limitações de recursos, menos flexibilidade com aplicativos complexos.
- 2) O Linux possui um núcleo monolítico, não um micronúcleo, porque a maioria das funcionalidades do sistema operacional, incluindo drivers e sistemas de arquivos, são executadas no espaço do núcleo em vez de em espaço de usuário separado. Isso resulta em maior eficiência de desempenho, mas menor modularidade em comparação com sistemas baseados em micronúcleo.
- 3)

Respostas resumidas:

 - (a) Incorreta. Máquinas virtuais de sistema não são específicas para uma linguagem, mas para sistemas operacionais.
 - (b) Incorreta. Um hipervisor hospeda máquinas virtuais, não sistemas operacionais hospedeiros.
 - (c) Correta. Em um micronúcleo, os componentes são módulos executando dentro do núcleo.
 - (d) Incorreta. Núcleos monolíticos não são conhecidos por sua facilidade de manutenção devido à complexidade.
 - (e) Correta. Em um micronúcleo, chamadas de sistema usam trocas de mensagens.

Exercício Cap 4.

- 1) Time sharing é um conceito em sistemas operacionais que permite que vários usuários compartilhem recursos de um computador, como CPU e memória, simultaneamente. Sua importância reside na eficiência no uso de recursos, multitarefa, economia de custos e produtividade melhorada.
- 2) A duração de um quantum de processamento é escolhida com base em vários critérios, incluindo o tipo de sistema operacional, o hardware disponível, os objetivos de desempenho, a carga de trabalho, a política de agendamento e a compatibilidade de aplicativos. Não há uma única resposta correta, e a escolha varia de sistema para sistema. Em geral, a duração do quantum afeta a responsividade e o desempenho do sistema, e deve ser ajustada para atender às necessidades específicas de cada caso.
- 3)
 1. E → P (Executando para Pronta): Possível, quando uma tarefa em execução é pausada e colocada na fila de tarefas prontas.
 2. E → S (Executando para Suspensa): Possível, quando uma tarefa em execução é temporariamente suspensa.
 3. S → E (Suspensa para Executando): Possível, quando uma tarefa suspensa é retomada.
 4. P → N (Pronta para Nova): Possível, quando uma tarefa pronta é retirada antes da execução e retornada ao estado de nova.
 5. S → T (Suspensa para Terminada): Possível, quando uma tarefa suspensa é encerrada prematuramente.
 6. E → T (Executando para Terminada): Possível, quando uma tarefa em execução é concluída.
 7. N → S (Nova para Suspensa): Possível, quando uma tarefa recém-criada é suspensa imediatamente após a criação.
 8. P → S (Pronta para Suspensa): Possível, quando uma tarefa pronta é suspensa antes da execução.
- 5)
 - [N] O código da tarefa está sendo carregado.
 - [P] A tarefas são ordenadas por prioridades.
 - [E] A tarefa sai deste estado ao solicitar uma operação de entrada/saída.
 - [T] Os recursos usados pela tarefa são devolvidos ao sistema.
 - [P] A tarefa vai a este estado ao terminar seu quantum.
 - [E] A tarefa só precisa do processador para poder executar.
 - [S] O acesso a um semáforo em uso pode levar a tarefa a este estado.
 - [E] A tarefa pode criar novas tarefas.
 - [E] Há uma tarefa neste estado para cada processador do sistema.
 - [S] A tarefa aguarda a ocorrência de um evento externo.