

Atividade de Sistemas Operacionais

Parte 2 (10 Questões)

Questão 1 (Capítulo 20 – Softwares de entrada e saída)

Os sistemas operacionais exigem que todos os drivers de dispositivo tenham a mesma interface pois será necessário que esses drivers funcionem de "forma universal", para que o sistema operacional não tenha que saber todas as peculiaridades de cada dispositivo instalado.

Questão 3 (Capítulo 21 – Discos rígidos)

A - RAID 0.

B - RAID 1: neste nível, o conteúdo é replicado em dois ou mais discos, sendo por isso comumente chamado de espelhamento de discos. A Figura 21.5 mostra uma configuração usual deste nível de RAID, com dois discos físicos (embora não seja usual, o conteúdo pode ser replicado em N discos, para tolerar N – 1 falhas).

C - RAID 0: O maior espalhamento dos blocos sobre os discos físicos contribui para distribuir melhor a carga de acessos entre eles e assim ter um melhor desempenho.

D - RAID 0: O maior espalhamento dos blocos sobre os discos físicos contribui para distribuir melhor a carga de acessos entre eles e assim ter um melhor desempenho.

E - RAID 5: assim como o RAID 4, esta abordagem também armazena informações de paridade para tolerar falhas em blocos ou discos. Todavia, essas informações não ficam concentradas em um único disco físico, sendo distribuídas igualmente entre eles.

Questão 3 (Capítulo 22 – O conceito de arquivo)

Temos a extensão por nome e os números mágicos.

→ Extensão por nome: Nesse tipo de abordagem a atribuição do tipo é feita através do próprio nome do arquivo.

→ Números Mágicos (Magic Numbers): Nesse tipo de abordagem a atribuição é feita em nível de bytes, onde alguns dos bytes no início do arquivo são selecionados para definir o formato.

Questão 4 (Capítulo 22 – O conceito de arquivo)

A - Magic Number são os bytes iniciais usados para identificar um arquivo.

ERRADO

B - CORRETO

C - CORRETO

D - CORRETO

E - ELF e PE são arquivos de código no sistema UNIX. **ERRADO**

F - MIME é o tipo para arquivos que serão enviados por email. **ERRADO**

Questão 3 (Capítulo 23 – Uso de arquivos)

→ **Acesso Sequencial:** nessa forma de acesso os arquivos são lidos em sequência do início ao fim do arquivo.

→ **Acesso aleatório:** no método de acesso aleatório (ou direto), pode-se indicar a posição no arquivo onde cada leitura ou escrita deve ocorrer, sem a necessidade de um ponteiro de posição corrente. Assim, caso se conheça previamente a posição de um determinado dado no arquivo, não há necessidade de percorrê-lo sequencialmente até encontrar o dado desejado. Essa forma de acesso é muito importante em gerenciadores de bancos de dados e aplicações congêneres, que precisam acessar rapidamente as posições do arquivo correspondentes aos registros desejados em uma operação. Na prática, a maioria dos sistemas operacionais usa o acesso sequencial como modo básico de operação, mas oferece operações para mudar a posição do ponteiro de acesso do arquivo caso necessário, o que permite então o acesso direto a qualquer registro do arquivo.

→ **Acesso mapeado em memória:** esse é uma forma particular de acesso aleatório onde mecanismos de paginação são utilizados. Nessa modalidade o arquivo é associado a um vetor de bytes do tamanho da memória principal, e cada posição do vetor corresponde à sua posição equivalente no arquivo.

→ **Acesso indexado:** nesta forma de acesso os arquivos são implementados com uma estrutura interna na forma de uma tabela de pares chave/valor. Os dados são salvos em registros com chaves associados a eles, para recuperá-los as chaves são utilizadas. Processos que estejam envolvidos no acesso a um mesmo arquivo devem travá-lo explicitamente no momento do acesso.

Questão 4 (Capítulo 23 – Uso de arquivos)

→ **Travas obrigatórias** - (mandatory locks): são impostas pelo núcleo de forma incontornável: se um processo obtiver a trava de um arquivo, outros processos que solicitarem acesso ao mesmo arquivo serão suspensos até que aquela trava seja liberada. É o tipo de trava mais usual em sistemas Windows.

→ **Travas recomendadas** - (advisory locks): não são impostas pelo núcleo do sistema operacional, mas gerenciadas pelo suporte de execução (bibliotecas). Os processos envolvidos no acesso aos mesmos arquivos devem travá-los explicitamente quando forem acessá-los. Contudo, um processo pode ignorar essa regra e acessar um arquivo ignorando a trava, caso necessário. Travas recomendadas são úteis para gerenciar concorrência entre processos de uma mesma aplicação. Neste caso, cabe ao programador implementar os controles necessários nos processos para impedir acessos conflitantes aos arquivos compartilhados.

→ **Travas exclusivas** - (ou travas de escrita): garantem acesso exclusivo ao arquivo: enquanto uma trava exclusiva estiver ativa, nenhum outro processo poderá obter outra trava sobre o mesmo arquivo.

→ **Travas compartilhadas** - (ou travas de leitura): impedem outros processos de criar travas exclusivas sobre aquele arquivo, mas permitem a existência de outras travas compartilhadas.

Questão 5 (Capítulo 23 – Uso de arquivos)

- **Semântica imutável:** de acordo com esta semântica, se um arquivo pode ser compartilhado por vários processos, ele é marcado como imutável, ou seja, seu conteúdo somente pode ser lido e não pode ser modificado. É a forma mais simples de garantir a consistência do conteúdo do arquivo entre os processos que compartilham seu acesso, sendo por isso usada em alguns sistemas de arquivos distribuídos.
- **Semântica UNIX:** toda modificação em um arquivo é imediatamente visível a todos os processos que mantêm aquele arquivo aberto.
- **Semântica de sessão:** considera que cada processo usa um arquivo em uma sessão, que inicia com a abertura do arquivo e que termina com seu fechamento.
- **Semântica de transação:** uma transação é uma sequência de operações de leitura e escrita em um ou mais arquivos, emitidas por um processo e delimitadas por comandos de início e fim de transação (begin ... end), como em um sistema de bancos de dados.

Questão 1 (Capítulo 24 – Sistemas de arquivos)

- **Dispositivos:** como discos rígidos ou bancos de memória flash, são os responsáveis pelo armazenamento de dados.
- **Controladores:** são os circuitos eletrônicos dedicados ao controle dos dispositivos físicos. Eles são acessados através de portas de entrada/saída, interrupções e canais de acesso direto à memória (DMA).
- **Drivers:** interagem com os controladores de dispositivos para configurá-los e realizar as transferências de dados entre o sistema operacional e os dispositivos. Como cada controlador define sua própria interface, também possui um driver específico.
- **Gerência de blocos:** gerencia o fluxo de blocos de dados entre as camadas superiores e os dispositivos de armazenamento. Como os discos são dispositivos orientados a blocos, as operações de leitura e escrita de dados são sempre feitas com blocos de dados, e nunca com bytes individuais.
- **Alocação de arquivos:** realiza a alocação dos arquivos sobre os blocos lógicos oferecidos pela camada de gerência de blocos. Cada arquivo é visto como uma sequência de blocos lógicos que deve ser armazenada nos blocos dos dispositivos de forma eficiente, robusta e flexível.
- **Sistema de arquivos virtual:** o VFS (Virtual File System) constrói as abstrações de diretórios e atalhos, além de gerenciar as permissões associadas aos arquivos e as travas de acesso compartilhado. Outra responsabilidade importante desta camada é manter o registro de cada arquivo aberto pelos processos, como a posição da última operação no arquivo, o modo de abertura usado e o número de processos que estão usando o arquivo.
- **Interface do sistema de arquivos:** conjunto de chamadas de sistema oferecidas aos processos do espaço de usuários para a criação e manipulação de arquivos.

→ **Bibliotecas de entrada/saída:** usam as chamadas de sistema da interface do núcleo para construir funções padronizadas de acesso a arquivos para cada linguagem de programação.

Questão 18 (Capítulo 24 – Sistemas de arquivos)

Na abordagem de mapa de bits, um pequeno conjunto de blocos na área reservada do volume é usado para manter um mapa de bits. Nesse mapa, cada bit representa um bloco lógico da partição, que pode estar livre ou ocupado. O mapa de bits tem como vantagens ser simples de implementar e ser bem compacto: em um disco de 500 GBytes com blocos lógicos de 4.096 bytes, seriam necessários 131.072.000 bits no mapa, o que representa 16.384.000 bytes, ocupando 4.000 blocos (ou seja, 0,003% do total de blocos lógicos do disco).

Questão 2 (Capítulo 25 - Diretórios e atalhos)

O layout conhecido como Sistema de Arquivos Unificado, do Linux, nos permite enumerar algumas das vantagens sobre o modelo utilizado pelo Microsoft Windows. Um exemplo é a pasta /usr do Linux. Esta pasta é utilizada para armazenar alguns executáveis utilizados na administração do sistema. Um outro ponto que difere é a troca da famosa contra barra do **Windows** "\" pela barra comum "/" nos sistemas **UNIX**.