|  |  |
| --- | --- |
|  | **FIPP – SISTEMAS DE INFORMAÇÃO – PRESIDENTE PRUDENTE.** |

**ANDRESSA HISAE TSUKASAKI – R.A. 261742078  
GABRIEL MIGUEL NAVAS – R.A. 261741888  
GUSTAVO DILLIO LEITÃO – R.A. 261741748**

**QUESTÕES REFERENTE A GERENCIAMENTO DE ARQUIVO**

Presidente Prudente - SP

2019

Questões referente a Gerenciamento de Arquivo

1. **Quais são os três requisitos fundamentais quanto falamos do problema do armazenamento de informações a longo prazo. Explique cada um deles.**

* Possibilidade de armazenar grande quantidade de informação: o processo possui suas informações dentro do seu espaço de endereçamento, ficando restrita ao tamanho do espaço virtual (para algumas aplicações é suficiente, para outros, tipo banco, não é)
* A informação (dentro do espaço de endereçamento do processo) deve sobreviver ao término do processo: em caso de algum acontecimento e o processo se perder, a informação deve ficar intacta. Dependendo da instituição, deve sobreviver por anos.
* Processos devem ser capazes de manipular concorrentemente informações ao mesmo tempo: um problema é que as informações retidas dentro do espaço de endereçamento de um processo não estão disponíveis para outros acessarem as informações, a solução é tomar a informação independente de qualquer processo.

1. **Explique quais são as Estruturas interna de Arquivos e quais são as formas de Acesso Interno a Arquivos, comentando as vantagens e desvantagens de cada uma.**

**Estruturas internas**

* **Sequência de bytes**: não tem estrutura lógica, Unix e Windows implementam e possui maior flexibilidade (colocamos qualquer coisa sem interferência do SO)
* **Sequência de registros de tamanho fixo**: cada um possui uma estrutura interna, seu objetivo é a leitura e gravação de registros no lugar de bytes.
* **Árvore de registros**: ordenada por campo chave, os registros não têm o mesmo tamanho, mais utilizado em computadores de grande porte (processamento de dados comerciais).

**Formas de acesso**

* **Acesso sequencial**: lê todos os Bytes/registros partindo do início, não era possível saltar arquivos, usados em fitas magnéticas e nos primeiros SO.
* **Acesso aleatório**: lê em qualquer ordem sendo movendo o marcador (seek) e dando o read, ou lê e move o marcador, é essencial para sistemas de base de dados.

1. **Enumere os principais atributos de um arquivo**.
2. Proteção - Quem tem acesso ao arquivo e de que modo
3. Senha - Necessidade de senha para acesso ao arquivo
4. Criador - ID do criador do arquivo
5. Proprietário - Proprietário atual
6. Flag de somente leitura - 0 para leitura/escrita; 1 para somente leitura
7. Flag de oculto - 0 para normal; 1 para não exibir o arquivo
8. Flag de sistema - 0 para arquivos normais; 1 para arquivos de sistema
9. Flag de arquivamento - 0 para arquivos com backup; 1 para arquivos sem backup
10. Flag de ASCII/binário - 0 para arquivos ASCII; 1 para arquivos binários
11. Flag de acesso aleatório - 0 para acesso somente sequencial; 1 para acesso aleatório
12. Flag de temporário - 0 para normal; 1 para apagar o arquivo ao sair do processo
13. Flag de travamento - 0 para destravados; diferente de 0 para travados
14. Tamanho do registro - Número de bytes em um registro
15. Posição da chave - Posição da chave em cada registro
16. Tamanho da chave - Número de bytes na chave
17. Momento de criação - Data e hora de criação do arquivo
18. Momento do último acesso - Data e hora do último acesso do arquivo
19. Momento da última alteração - Data e hora da última modificação do arquivo
20. Tamanho atual - Número de bytes no arquivo
21. Tamanho máximo - Número máximo de bytes no arquivo
22. **Enumere as principais operações sobre arquivos.**
23. Create: Criado um arquivo sem dados.
24. Delete: Deleta um arquivo, liberando espaço em disco.
25. Open: Abrir um arquivo antes de utiliza-lo.
26. Close: Usado para fechar um arquivo aberto liberando espaço na tabela interna (atributos e endereços do disco)
27. Read: Usado para ler de um arquivo aberto, da posição atual a quantidade de bytes especificada colocando num buffer também especificado.
28. Write: Escreve em posição determinada (subscreve se for no meio) e adiciona se for no final.
29. Append: Adiciona dados no final do arquivo.
30. Seek: Reposiciona o ponteiro do arquivo.
31. Get Attributes: Retorna atributos do arquivo.
32. Set Attributes: Altera os dados do arquivo.
33. Rename: Renomeia o arquivo já existente.
34. **O que é um descritor de arquivo? Para que ele serve?**  
     São variáveis declaras no momento de criação ou abertura de um arquivo. São usadas para identificar se o arquivo foi aberto corretamente e usadas na maioria das funções em arquivo, como read (leitura), write (escrita) e open (abrir).
35. **Explique os 3 tipos de níveis de diretórios existentes.**

**1. Sistemas de diretório em nível único**: É a forma mais simples de sistema de diretório, pois existe somente a raiz, através desta contém todos os arquivos, existindo somente um lugar para procurar (a raiz) arquivos. Usado nos primórdios da computação, também em alguns dispositivos embarcados simples como câmera digital e alguns tocadores portáveis de música. Pode-se achar arquivos rapidamente.

**2. Sistema de diretório em dois níveis**: Cada usuário tem seu diretório pessoal, no segundo nível existem diretórios que são indexados pelo nome do usuário ou arquivos, já no terceiro nível existem os arquivos. Neste tipo de sistema permite usuários com mesmo nome, mas com isolamento entre eles.  
  
 **3. Sistemas de diretórios hierárquicos**: Utiliza uma hierarquia (isto é, uma árvore de diretórios), podendo navegar em diretórios ou subdiretórios e criar quantos forem necessários. Sendo possível também múltiplos usuários utilizar um servidor de arquivos por exemplo, utilizando esse sistema. Forma mais utilizada atualmente em ambos sistemas UNIX e Windows.

1. **Explique as diferenças entre caminho absoluto e caminho relativo.**  
   Caminho absoluto é quando se identifica o diretório atual ou caminho de arquivo desde sua raiz. exemplo: */home/navas/Desktop/foto\_ferias.jpg*

Caminho relativo é quando eu estou no diretório */home/navas/Desktop/* e quero acessar o diretório *mais\_fotos* eu não preciso passar o caminho todo, apenas indico a pasta que quero ir. Exemplo: cd mais\_fotos

A diferença é que se o diretório desejado estiver a frente ou atrás é só preciso passar o diretório, não o caminho todo.

1. **O que é fragmentação interna e externa referente a arquivos? Por que elas ocorrem?**

Fragmentação interna: Ao alocar um arquivo não se utiliza todo o espaço do bloco sobrando espaço dentro do mesmo, isso pode ocorrer, porque o arquivo é menor que o tamanho do bloco ou devido a divisão do arquivo em mais de um bloco o último bloco não completamente utilizado, exemplo: sistema utiliza blocos de 4 Bytes, e ao alocar um arquivo de 3 Bytes sobra-se 1 Byte não alocado dentro do bloco.

Fragmentação externa: Quando um arquivo é excluído ou movido, ele libera um bloco de memória do mesmo tamanho que o arquivo. Se o próximo arquivo salvo na medida em que o espaço é muito pequeno para preencher a memória disponível, ele vai deixar uma parte não utilizada que muitas vezes é pequena demais para qualquer outra coisa para ser salvo dentro. Logo, é alocado em outra posição do disco.

1. **Explique o que é alocação contígua de arquivos, apresentando suas vantagens e desvantagens.**

Uma implementação de arquivos na qual os arquivos ficam guardados sequencialmente em blocos, quando o arquivo é removido os blocos ficam vazios. Possui as vantagens de ser simples de implementar (deve-se lembrar de apenas dois números, endereço em disco do primeiro bloco e número de blocos do arquivo), o desempenho de leitura é excelente (todo arquivo pode ser lido em uma única operação, só colocar um seek e realizar a leitura). Suas desvantagens são que o disco fica fragmentado com o tempo, custa para fazer a realocação dos blocos dos arquivos, utilizadas em CD e DVD.

1. **Explique a alocação de arquivos em listas ligadas no bloco, apresentando suas principais vantagens e desvantagens.**

Uma implementação de arquivos na qual partes do arquivo ficam em diversos blocos um apontando para o endereço do outro dentro do disco. Suas vantagens são que não há fragmentação (somente no último) e mantém uma única entrada de diretório para o endereço do primeiro bloco. Suas desvantagens são que o acesso aleatório é lento (para chegar em N tem que passar pelos anteriores), quantidade de dados não é potência de 2 (programas leem potência de 2), deve-se concatenar dois blocos e isso gera sobrecarga.

1. **Explique a estrutura do sistema de arquivos conhecido como FAT, comentando sobre suas qualidades e deficiências.**

Tabela em que cada bloco possui o endereço do outro bloco em que está a outra parte do arquivo. Suas vantagens são que elimina o ponteiro do bloco e o acesso é rápido. Suas desvantagens são que a tabela inteira deve estar na memória e possui entradas para todos os blocos de dados do disco.

1. **Explique a estrutura do sistema de arquivos conhecido como I-Node, comentando sobre suas qualidades e deficiências.**

Cada I-Node aponta para um arquivo, ou seja, cada bloco da tabela do i-node aponta para um ou mais blocos de um arquivo. Cada arquivo no disco tem o seu próprio i-node. Suas vantagens são que não precisa ficar na memória inteiramente, apenas quando o arquivo está aberto. Suas desvantagens são que pode não ter espaço naquele determinado endereço para inserir o arquivo (a solução é reservar o último endereço para o endereço de um bloco que contém mais endereços de blocos de disco).

1. **Analise o impacto das fragmentações interna e externa nos sistemas de alocação contígua, lista encadeada e por lista encadeada na RAM.**

* **Alocação contígua**: Pode haver ambas fragmentações (desperdício de blocos ou falta deles) devido às exclusões dos dados que deixam os blocos vazios. Para inserir outros dados nesses blocos pode acontecer as fragmentações. Apesar da rapidez de acesso, é necessária uma reorganização com frequência para eliminar os blocos livres e quanto mais se acessa o disco mais se desgasta.
* **Lista Encadeada**: Não possui fragmentação externa, pois em cada bloco é guardado o ponteiro que aponta para o próximo, mas no último bloco está o “null” que anuncia o final das ligações e este é o único que possui fragmentação interna.
* **Lista Encadeada na RAM**: Semelhante à lista encadeada, apesar de ser uma tabela, a única fragmentação é a interna no último bloco que informa o final das ligações.

1. **Descreva e analise as seguintes técnicas de alocação de blocos de arquivos em discos magnéticos: (a) Alocação Contígua; (b) lista encadeada; e (c) lista encadeada em RAM.**

**Alocação contígua:** De forma contígua, é alocado blocos desde o início do disco, cada grupo de blocos é um arquivo. Se tiver 3 arquivos, o primeiro grupo será o primeiro o arquivo, o segundo será o 2º grupo de blocos e o 3º grupo será o terceiro arquivo, tendo um total de 3 arquivos em 3 grupos sequenciais de blocos, cada grupo de blocos tendo seus números de arquivos de tamanhos iguais ou não.

**Lista Encadeada:** Nesta lista, cada bloco tem um ponteiro que aponta para o próximo bloco, dentro do bloco, o resto dele, é reservado para os dados, assim sucessivamente.

**Lista Encadeada em RAM:** Neste tipo é quase igual à lista encadeada a cima, só que podem ser eliminadas colocando-se as palavras do ponteiro de cada bloco de disco em uma tabela na memória principal. Usando essa organização, o bloco inteiro fica disponível para dados. Além disso, o acesso aleatório é muito mais fácil. Embora ainda seja necessário seguir o encadeamento para encontrar um determinado deslocamento dentro do arquivo, o encadeamento está inteiramente na memória, portanto ele pode ser seguido sem fazer quaisquer referências ao disco.