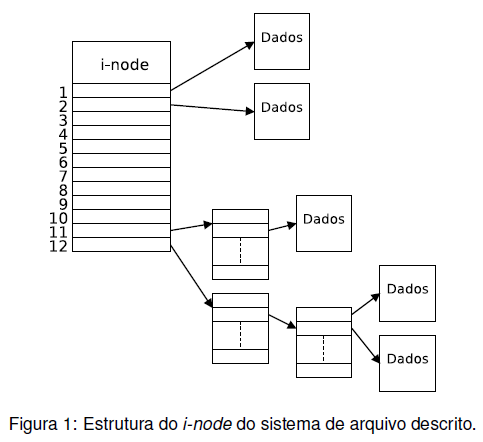
**783Z Sistemas Operacionais Abertos - Lista de Exercícios 2**

**Prof.: Renê de Souza Pinto**

**Questão 1**: Considere um sistema de arquivo baseado em i-nodes, que utiliza blocos de dados de 1KB e 32 bits para o endereçamento de cada bloco. A figura 1 mostra a estrutura de cada i-node, que possui 12 campos que apontam para blocos, sendo 10 apontamentos diretos, um indireto simples e um indireto duplo. Qual é o máximo tamanho de arquivo permitido neste sistema?



**Questão 2**: Considere um sistema de arquivos baseado em i-node. Qual a diferença entre links simbólicos e hard links?

R: Links simbólicos são uma solução para arquivos compartilhados onde um arquivo B qlq se liga a um outro arquivo C, isso faz com que o sistema crie um novo arquivo do tipo LINK e inclui este arquivo no diretório de B. Este novo arquivo contém apenas o nome do caminho do arquivo a qual está ligado. O HARD-LINK ao invés de criar um novo arquivo do tipo LINK com o caminho do arquivo, faz uma referencia ao mesmo i-node do arquivo original, desta forma ele se manterá intacto mesmo que o arquivo original sofra modificações, enquanto uma ligação simbólica se o arquivo original sofrer alterações ou for movido de pasta o arquivo do tipo LINK ficará corrompido

**Questão 3**: Explique como funciona Mapa de Bits, utilizado por diversos sistemas de arquivos para o controle de blocos livres e ocupados.

R: Com um mapa de bits, a memória é dividida em unidades de alocação, que podem conter poucas palavras ou ter vários quilobytes. Associado a cada unidade de alocação existe um bit no mapa de bits, o qual vale 0 se se a respectiva unidade de alocação estiver disponível e 1 se estiver ocupada.

**Questão 4:** Cite e explique pelo menos dois tipos de inconsistência que um desligamento abrupto do sistema pode ocasionar ao sistema de arquivo.

R: O desligamento abrupto do sistema pode causar perda de arquivos ou corrupção dos mesmos. Imagine que se toda vez que você for precisar salvar algum dado ter que mover o braço de escrita/leitura do disco rígido? Isso demandaria tempo, por isso que as informações são alocadas em blocos na memória antes de serem gravadas no disco, minimizando o tempo caso o sistema seja interrompido, as informações em blocos que ainda não foram salvas serão perdidas. Também pode ocorrer de corromper uma determinada aplicação que não teve todos os dados salvos corretamente devido a essa perda.

**Questão 5:** Quais as vantagens do cache de blocos de disco mantido pelo kernel? Qual(is) problema(s) o uso deste cache pode ocasionar?

R: Um cache de bloco de disco é utilizado para acelerar o acesso aos dados que estão sendo mais frequentemente requeridos. Quando um programa precisa acessar um novo dado, o SO primeiramente verifica se os dados estão no cache antes de lê-lo do disco já que a acesso em memória ram é muito mais rápida do que o acesso ao disco. Caso aconteça uma interrupção brusca no sistema pode ocorrer perda de informações contidas no cache de blocos por estas estarem na memória ram.

**Questão 6:** Em relação ao kernel Linux, responda:

**a)** Qual é o tipo de arquitetura do kernel?

R: Arquitetura Monolítica

**b)** Cite uma vantagem e uma desvantagem desta arquitetura

R: O ponto posítivo de um kernel monolítico é que ele proporciona melhor segurança e melhor desempenho devido seus recursos residirem dentro do próprio kernel (built-in), por outro lado o mal funcionamento de uma aplicação do kernel pode se alastrar levar todo o sistema a um colapso.

**c)** Explique detalhadamente o funcionamento do kernel através desta arquitetura

R: Em um sistema monolítico, todos os componentes do núcleo operam em modo núcleo e se inter-relacionam conforme suas necessidades, sem restrições de acesso entre si (pois o código no nível de núcleo tem acesso pleno a todos os recursos e áreas de memória).

**d)** Qual é o mecanismo de comunicação entre o kernel e os aplicativos do espaço de usuário?

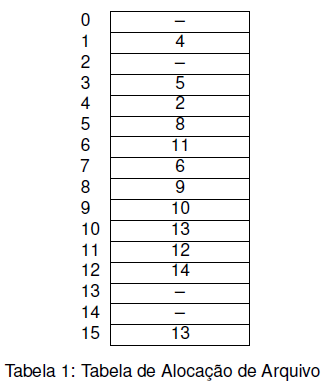
R: A comunicação de um aplicativo de usuário com o núcleo do sistema é feita via chamadas ao sistema (System Calls), essas chamadas dizem o que realmente um SO faz, no caso do UNIX temos quatro tipos de chamadas ao sistema, o primeiro relaciona-se com a criação e finalização de processos, o segundo grupo é para leitura e escrita em arquivos, o terceiro é voltado ao gerenciamento de diretórios e o quarto grupo contém chamadas diversas.

**Questão 7:** Explique o que é e quais as diferenças entre Espaço de Kernel e Espaço de Usuário.

R: No espaço de Kernel temos funções e programas do sistema que gerenciam todo o hardware, possui uma estrutura complexa que determina como os recursos devem ser gerenciados, é a parte mais interna de um S.O e é protegida dos usuário pelo hardware.

No Espaço de usuário encontramos os programas de usuários como compiladores e editores de textos por exemplo. Se um usuário não gostar de um determinado compilador ele poderá escrever seu próprio compilador, mas não lhe é permitido escrever sua própria rotina de interrupções de relógio, que é parte do SO, e está protegida pelo hardware contra tentativas de alteração do usuário.

**Questão 8:** Considere a Tabela 1, que representa a Tabela de Alocação de Arquivo de um determinado sistema de arquivos que utiliza blocos de dados de 4KB.



Sabendo que o arquivo Foto001.jpg começa no bloco 3 (índice 3 na tabela), responda:

**a)** Como funciona a Tabela de Alocação de Arquivo?

R: O armazenamento em uma tabela de alocação de arquivo é feita como uma lista encadeada de blocos de disco. A primeira palavra de cada bloco é usada como um ponteiro para um próximo, ambos os encadeamento tem uma marca de término (por exemplo, -1) que corresponde a um número inválido de bloco.

**b)** Quais blocos pertencem ao arquivo?

**c)** Qual o tamanho total do arquivo?

**Questão 9:** Explique resumidamente os passos para a configuração e compilação do kernel Linux.

R: - baixar código fonte;

- Utilizar interfaces de configuração ou editar via *.config*;

- Executa *make* para compilar;

- Instalação do kernel

* Gerar pacote *.debian* (Debian) ou *.rpm* (Fedora);
* *Make install* / *make mode install* / *make firmware*

**Questão 10:** Comente a vantagem da implementação de drivers de dispositivo através de módulos no Linux.

R: Economizar memória e processamento permitindo que os drivers fiquem carregados no sistema de arquivos e só consumam memória quando carregados/ instalados (suporte a módulos)

**Questão 11:** Cite alguns comandos de um ambiente Linux que podem ser utilizados para obter informações a respeito do hardware da máquina na qual o SO está sendo executado.

R: O comado **“free”** pode ser utilizado para obter informações da memória física e **swap** disponíveis no sistema, este comando pega as informações do do arquivo /**proc/meminfo**.

O comando **“df”** *(disk free)* mostra o espaço livre/ocupado de cada partição.

O comando **cat** **proc/cpuinfo** – Este arquivo possui detalhes sobre a nomenclatura do processador, exibe por exemplo, a quantidade de núcleos, frequência, família de processador.

**Questão 12:** Em relação a configuração para compilação do kernel Linux, explique sucintamente o que são as opcções:

* Módulo – drive complilado em formato de módulos, ou seja, você carrega somente os que for utilizar
* Built-in – dentro do kernel, ou autoconstruído. Significa que o driver é carregado/ compilado junto com instalação do kernel
* Sem suporte – kernel não terá suporte àquele hardware – não será compilado

**Questão 13:** Explique como funciona de maneira geral o suporte a módulos do kernel Linux. O código de cada módulo é executado em Espaço de Kernel ou Espaço de Usuário?

R: Os módulos são funcionalidades (código) que estão incluídos originalmente dentro do kernel, podem estar em disco de maneira inativa e passa a atuar na memória principal junto ao kernel (no espaço de Kernel) assim que o habilitamos. Proporcionam a vantagem de executar somente o que for necessário deixando um kernel enxuto, isso faz com que mantenha os recursos computacionais mais livres (como uso de memória e processamento) caso não haja necessidade de utilizá-los.

**Questão 14:** Cite e comente sobre pelo menos dois comandos relacionados a manipulação de módulos do kernel Linux.

R: lsmod: esse comando tem a função de listra todos os módulos que estão ativos no sistema, mesmo que o módulo não esteja sendo usado no momento.

modprobe nome\_do\_módulo: ele é o responsável por ativar um módulo, é através dele que habilitamos um determinado dispositivo, como por exemplo a placa de rede.

**Questão 15:** Comente sucintamente sobre os comandos:

* modprobe – busca módulos no diretório padrão e carrega na memória. Faz também a checagem de dependência entre os módulos
* insmod – carrega na memória o módulo que for “descer”
* rmmod – remove o módulo carregado na memória
* modinfo – mostra informações sobre determinado módulo
* lsmod – lista os módulos carregados no sistema
* dipmode – listagem das dependências entre módulos

**Questão 16:** O que é um firmware?

R: Também conhecidos como “software embarcado” é o conjunto de instruções operacionais programadas diretamente no hardware de um equipamento eletrônico. É armazenado permanentemente num circuito integrado (chip) de memória de hardware, como uma ROM, PROM, EPROM ou ainda EEPROM e memória flash, no momento da fabricação do componente.

**Questão 17:** Considerando a evolução do hardware embarcado na ultima década e o surgimento de plataformas de considerável poder computacional e baixo custo (tais como a placa Raspberry PI), comente a importância de Sistemas Operacionais abertos como o Linux, para que se possa explorar de forma otimizada todos os recursos destas plataformas.

(pergunta de resposta aberta)

R: É fundamental para desenvolver soluções a essas plataformas, através do conhecimento de um kernel robusto e a implementação personalizável que se adapta as necessidades para a aplicação final.

Projetos de portabilidade de novos dispositivos são mais fáceis de serem desenvolvidos, pois o código fonte estão disponíveis e facilitam no desenvolvimento de soluções para instalação e uso de determinados dispositivos.