

Apostila
Introdutória
de Linux
Ver. 4.1

Materiais de Apoio às Aulas
práticas em Laboratório

Prof. Luís Rodrigo de O. Gonçalves
luisrodrigoo@yahooc.com.br
<http://www.Incc.br/~lrodrigo>

- 1) Apresentação do Curso
- 2) Histórico do Unix e Linux
- 3) Instalação do Linux
- 4) Estrutura de Diretórios do Linux
- 5) Conceitos Básicos
- 6) Comandos Básicos
 - 6.1) Entrando e Encerrando o Sistema
 - 6.2) Gerenciamento de Diretórios
 - 6.2.1) Listando Arquivos
 - 6.2.2) Meta Caracteres
 - 6.2.3) Criando, Removendo e Listando Diretórios
 - 6.2.4) Copiando, Removendo, Movendo e Renomeando Arquivos
 - 6.2.5) Mostrando o conteúdo de arquivos
 - 6.2.6) Links
 - 6.2.7) Comparando Arquivos
 - 6.3) Pedindo ajuda
 - 6.4) Permissão dos Arquivos
 - 6.5) Pipe e Redirecionadores
 - 6.6) Editores de Texto
 - 6.7) Localizando Arquivos
 - 6.8) Manipulação de Texto
 - 6.9) Sistema de Arquivo
 - 6.10) Gerenciamento de Processos
 - 6.11) Compactação e Backup
 - 6.12) Data e Hora
 - 6.13) Comandos de Vídeo
- 7) Lista de Exercícios

1) Apresentação do Curso

1.1) Objetivos:

Apresentar, de forma prática, os conceitos básicos do sistema operacional Linux, através de aulas práticas no laboratório e quando necessárias aulas teóricas na própria sala de aula.

1.2) Referências:

1.2.1) Livros:

- 1.) Danesh, Arman; Dominando o Linux – Red Hat Linux 6.0, A Bíblia; Ed. Makron Books
- 2.) Welsh, Matt; Dominando o Linux; Ed. Ciência Moderna
- 3.) Siever, Ellen; Linux, O Guia Essencial; Editora Campus
- 4.) Petrovsjy, Michele e Parkinson, Tom; Guia de Referência do Unix; Quark Books

1.2.2) Sites da Internet

- 1.) Conectiva Linux – <http://www.conectiva.com.br>
- 2.) RedHat Linux – <http://www.redhat.com.br>
- 3.) Linux Security Brasil – <http://www.linuxsecurity.com.br>

2) *Histórico do Unix*

O primeiro sistema operacional Unix foi criado em meados de 1969 nos Laboratórios da BELL TELEPHONE (BTL). Ele foi inicialmente escrito utilizando-se a linguagem de programação Assembler, que lhe conferia maior velocidade, porém baixo nível de portabilidade. Com o surgimento da Linguagem de programação "C" o seu "cernelha" foi reescrito permitindo que o mesmo fosse portado para várias plataformas de hardware.

Desde o seu surgimento, o Unix tem ganhado cada vez mais espaço, não somente na comunidade acadêmica, como também no mercado. No final da década de 80 até meados da década de 90 ele perdeu parte de seu espaço de atuação para os sistemas operacionais da Microsoft, mas com o advento do Linux este processo tem sido invertido, e atualmente ele tem sido um dos sistemas operacionais mais utilizados.

Inicialmente desenvolvido para computadores de grande porte, mainframes, atualmente tem sido largamente utilizado em microcomputadores.

Um Sistema Unix é composto basicamente de uma coleção de aplicativos para os usuários, bibliotecas, utilitários e o próprio ambiente operacional que é o responsável por fazer a interface entre o hardware e o usuário final.

Em meados 1994 foi oficialmente iniciado o desenvolvimento de um sistema operacional baseado no UNIX porém que "rodava" em PC, na época em máquinas 386, este sistema ficaria conhecido posteriormente como Linux, sendo distribuído de forma gratuita.

2.1) *Características do Unix*

O UNIX, inicialmente, era um sistema operacional voltado para ambientes robustos, os quais geravam um elevado nível de serviço, os quais deveriam ser atendidos com alto nível de qualidade. Contudo com a migração que tem ocorrido nos últimos anos para ambientes menores o Unix continua atendendo as mesmas necessidades.

Dentre as suas principais características podemos destacar:

1. **Funcionalidade:** capacidade de se adaptar a novas necessidades a medida que elas são desenvolvidas.
2. **Suporte a Rede de Computadores:** dão suporte de forma nativa uma vasta gama de protocolos dos quais podemos destacar: TCP/IP, ATM, FDDI e etc; Assim como uma vasta gama de serviços de rede, tais como: NFS (Network File System), NIS (Network Information Service), E-mail, http, ftp, acesso remoto e etc...
3. **Performance:** tem sido considerado um dos sistemas operacionais mais rápidos do mercado, mesmo para ambiente de pequeno porte.
4. **Portabilidade:** suporta vários tipos de hardware e possui isolamento da camada de dependência.

5. **Qualidade / Disponibilidade:** melhoramento constante para o aumento da qualidade dos serviços; disponibilidade através de soluções de Cluster e Alta Disponibilidade.
6. Ainda pode-se destacar outras como: a segurança, a estabilidade, a capacidade de ser Multitarefa & Multiusuário, possibilidade de permitir a emulação de terminais, o acesso remoto, a Criação de Cluster e possui um padrão de regulamentação denominado Padrão “Posix”.

2.2) Estrutura do Unix – Shell x Kernel

O sistema operacional UNIX é dividido basicamente em duas camadas: kernel e shell. O Kernel o responsável pela interface entre o hardware e o shell, e é considerado como o “núcleo central” do sistema operacional. Acima deste é executado o shell que realiza a interface entre os aplicativos o usuário e o kernel. Existem uma série de versões de shell, porém, o mais utilizado no Linux é o BASH.

Devido a arquitetura do Unix um aplicativo pode rodar em dois modos, ou em modo kernel, no qual ele tem acesso direto aos dispositivos, ou em modo Usuário, no qual ele pede serviços à camada do kernel.

Objetivando facilitar a compreensão de desta estrutura, a seguir temos um diagrama exibindo como é estruturado o Unix padrão.

3.) Instalação do Linux

3.1) Primeiros passos para a Instalação do Linux

O processo de instalação do Linux pode ser dividido em vários passos, dos quais pode-se destacar os seguintes:

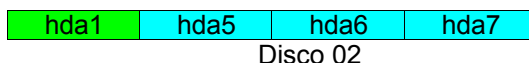
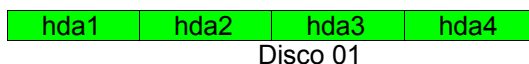
- Inventário do Hardware;
- Alocação espaço em disco para o Linux;
- Criação e ativação dos sistemas de arquivos;
- Seleção dos pacotes;
- Processo de instalação propriamente dito;
- Configuração do gerenciador de Boot (LILO).

3.1.1) Inventário:

Nesta etapa, o Linux irá realizar uma busca pelos hardwares instalados na máquina, bem como as suas configurações. Geralmente toda a placa ou dispositivo possuem associado a eles alguns valores que definem a sua configuração. Dentro destes valores, os mais importantes são o número de IRQ, de I/O e de DMA.

3.1.2) Alocando espaço em disco para o Linux e configuração do sistema de arquivo

Uma unidade de disco pode ser dividida em vários pedaços, cada um destes é chamado de Partição de Disco. Uma partição pode ser primária ou secundária, dependendo do seu tipo e da sua posição no disco ela recebe uma nomenclatura. A seguir tem-se a representação de dois discos rígidos do tipo IDE:



O Disco 01 é composto de 4 partições primárias, normalmente estas partições são utilizadas para a instalação de sistemas operacionais, já o Disco 02 possui uma partição primária e três secundárias, contudo um disco rígido pode conter quantas partições secundárias couberem no mesmo e não somente três.

Percebemos que mesmo os dois discos possuindo a mesma quantidade de partições a sua nomenclatura muda, o motivo da alteração da nomenclatura é a regra de formação utilizada para nomear as partições.

A seguir temos uma breve explicação desta regra utilizada na nomenclatura

- Se o disco for IDE ela começa com **hd**, se for SCSI ela começa com **sd**;
- De acordo com a posição do disco ela assume uma letra, como vemos na tabela abaixo:

1º Disco	hda	sda
2º Disco	hdb	sdb
3º Disco	hdc	sdc
4º Disco	hdd	sdd
Tipo	IDE	SCSI

- De acordo com a posição da partição dentro do disco ela assume um número, ou seja, se é uma das partições primárias este número vai de 1 até 4, porém, quando estamos trabalhando com uma partição secundária este número deve ser maior ou igual a 5.

Para a criação destas partições, existe uma vasta gama de ferramentas, dentre elas podemos destacar o “**fdisk**”, que mesmo não sendo um aplicativo simples sempre pode ser encontrado nas distribuições de Linux. Uma outra ferramenta disponível é o “fips” cujo objetivo é o *redimensionamento do disco, ou seja, alteração do tamanho das partições existentes, este também vem com o Linux*. Por outro lado existe uma ferramenta “paga” que é o “**Partitiom Magic**” que apresenta uma maior facilidade de uso. Maiores informações em www.powerquest.com.br

Porém, somente o processo de particionar um disco não permite que ele seja utilizado para a instalação do Linux, sendo necessário ainda que esta partição seja “formatada”. Quando formatamos uma determinada partição do disco estamos na realidade associando a esta partição um determinado sistema de arquivo. No nosso caso, os sistemas de arquivos mais importantes são:

- DOS/Windows:
 - FAT16
 - FAT32
 - VFAT
 - NTFS
- Linux
 - Ext2
 - Ext3
 - Linux Swap

Quando instalado o Linux pode estar instalado em várias partições, cada uma com um uso específico, porém duas partições são indispensáveis para o seu funcionamento:

/ - Raiz na qual todos os pacotes serão instalados

Swap - Usada como sendo uma região de disco que é utilizada como se fosse um pedaço da memória RAM; normalmente o tamanho desta partição é igual a duas vezes o tamanho da memória principal (RAM).

3.1.3) A seleção dos Pacotes

Nesta parte do processo de instalação serão selecionados quais aplicativos devem ser instalados. De acordo com o tipo de instalação a ser realizada existe um conjunto de pacotes padrão. A maioria das distribuições já possuem algumas pré-seleções de pacotes que o usuário pode escolher para facilitar a sua tarefa de instalação, evitando assim uma seleção manual.

Feita a seleção de pacotes, o processo de instalação e cópia dos pacotes para a partição e as configurações dos serviços são realizadas de forma quase que automática, ficando ao encargo do usuário apenas mais alguns detalhes que variam de distribuição para distribuição.

3.1.4) A configuração do Gerenciador de Boot.

Terminada a fase anterior temos que definir como o Linux será iniciado e caso haja outros sistemas operacionais instalados, como estes serão também carregados.

Existe no Linux um aplicativo denominado gerenciador de boot, que irá permitir escolher qual sistema será iniciado, bem como os parâmetros de inicialização, sendo possível a inserção de uma senha, a ser utilizada para evitar que a máquina seja acessada por algum usuário indevido.

O gerenciador de boot mais utilizado no Linux, tem sido, o "lilo" devido principalmente, a sua facilidade de configuração e a sua flexibilidade de uso. Mas o "lilo" pode apresentar problemas quando o mesmo estiver sendo utilizado em conjunto com alguns sistemas operacionais, como é o caso do OS/2 da IBM e o Windows NT da Microsoft, porém nestes casos existem alguns documentos na rede que podem ajudar a resolver estes problemas.

Uma outra opção ao "lilo" é o "Grub", que é um outro gerenciador de boot utilizado pelo Linux; mas este geralmente faz uso de uma interface gráfica, o que em alguns casos pode significar um certo problema.

4) Estrutura hierárquica dos diretórios no Linux

Os diretórios no Linux podem ser vistos como sendo uma árvore, só que de cabeça para baixo, ou seja, o primeiro diretório é o diretório '/', ou simplesmente Raiz, abaixo dele podem existir vários outros como o 'bin', 'sbin', 'etc', 'usr' e outros.

O motivo do uso de uma estrutura deste tipo é que em uma árvore somente existe um caminho desde a raiz até um determinado ponto da estrutura de diretório, o que evita o problema de se gerar um loop infinito quando da busca de um arquivo.

4.1) Principais diretórios do Linux

- /bin - utilitários principais;
- /dev - dispositivos, tais como: partições de disco, unidade de cdrom, fax-modem, etc;
- /etc - arquivos de configuração da máquina;
- /lib - bibliotecas e funções utilizadas por outros aplicativos;
- /tmp - arquivos temporários;
- /home - onde ficam localizados os diretórios pessoais dos usuários cadastrados;
- /usr/bin - outros utilitários;
- /usr/lib - bibliotecas de função;
- /usr/spool - spool de e-mail e impressora.

4.2) Path Absoluto e Relativo

O PATH é o caminho que descreve a localização de um objeto, seja ele um arquivo ou um diretório, dentro do sistema de arquivo, por exemplo: '/home/lrodrigo/aula3.txt'

Existem dois tipos de Path: o path absoluto e o path relativo. O path absoluto é tido como todo e qualquer caminho desde a raiz até o objeto desejado, um modelo deste tipo de PATH seria "/etc/fstab"

Já o Path relativo, é qualquer caminho que parte de um determinado ponto da árvore de diretório, que não seja a raiz, geralmente este ponto é o diretório corrente, por exemplo: supondo que atualmente estamos no diretório "/home" um PATH relativo, válido, poderia ser "lrodrigo/aula3.txt"5.)
Conceitos Básicos:

5.1.) Usuários

Existem dois grupos básicos de usuários :

- **Usuários comuns** – São aqueles usuários que podem utilizar os recursos do sistema dentro de limitações estabelecidas para eles, através da estrutura de proteção de arquivos.
- **Superusuários ou Administradores do Sistema** – Normalmente é um grupo reduzido de usuários, que podem acessar todo o sistema de maneira irrestrita, não estando sujeitos ao sistema de proteção de arquivos. Esta modalidade de utilização deve ser reservada exclusivamente para as atividades de Administração do Sistema.

5.2.) Ambiente do Usuário

Para um usuário acessar o sistema é necessário um procedimento de cadastramento, uma abertura de conta. Esta tarefa é realizada pelo Administrador do Sistema. Para isto é necessário obter dos usuários as seguintes informações :

- ✓ Seu nome e sugestão para o nome de **login**.
- ✓ Descrição de suas atividades e o relacionamento com outros usuários.
- ✓ Estimativa do espaço em disco necessário para execução de suas tarefas.

Com base nestas informações o Administrador do Sistema definirá os seguintes atributos para os usuários :

- ✓ Nome para se logar ao sistema (login);
- ✓ Senha de acesso (passwd);
- ✓ Identificação numérica do usuário (uid);
- ✓ Identificação numérica do grupo (gid);
- ✓ Diretório de trabalho (diretório home);
- ✓ Programa inicial a executar (normalmente /bin/sh).

Estas informações são armazenadas no diretório **/etc**, nos arquivos :

- ✓ **/etc/passwd** - Informações aos usuários
- ✓ **/etc/group** - Informações referentes aos grupos e seus membros

OBS: O sistema possui um único usuário com **uid 0** e um grupo com **gid 0**

O usuário com uid 0 é chamado de root ou superusuário e os usuários com gid 0 fazem parte do grupo de Administradores do Sistema. Estes possuem privilégios especiais que outros grupos não possuem. Todos os demais usuários devem ser cadastrados em outros grupos no sistema.

5.3) *Variáveis de Ambiente*

O sistema guarda um grupo de variáveis para formar o ambiente de trabalho do usuário, e, por convenção, estas variáveis tem seus nomes em letras maiúsculas. Abaixo relacionamos algumas delas.

PATH : Especifica os comandos disponíveis para um determinado usuário, pois guarda os nomes dos percursos permitidos na busca de comandos quando estes são digitados;

HOME : Guarda o nome do diretório de trabalho original do usuário (home);

MAIL : Guarda o nome da caixa postal do usuário para receber correspondência;

LOGNAME : Guarda o nome de identificação do usuário no processo de login;

TERM : Guarda o nome do tipo de terminal utilizado pelo usuário;

PS1 – Guarda o símbolo de prompt utilizado pelo usuário;

PS2 – Guarda o símbolo de prompt secundário. O default é **>**;

IFS – Guarda os separadores internos de campos. O default é espaço em branco.

6.) Comandos Básicos do Linux

Esta parte da apostila visa apresentar uma vasta gama de comandos largamente utilizados no Linux, o objetivo é apenas apresentar os comandos e não uma explicação completa sobre os mesmos

6.1.) Conectando-se e desconectando-se do sistema

Depois de iniciado, o Linux realiza um processo de validação do usuário que é executado através do login dos usuários. Toda vez que se deseja fazer uso de uma máquina Linux, deve realizar o processo de conexão, semelhante ao que fazemos quando usamos o caixa eletrônico dos bancos, porém no Linux fazemos uso de duas informações, a nossa identificação pessoal conhecida normalmente como "Login Name" e a senha ou "password". Para gerenciar este processo podemos fazer uso dos seguintes comandos:

Comando	login
Comentário	Utilizado para se conectar ao sistema, usando um login name válido
Sintaxe	Login [username] [username] é o nome do usuário o qual deseja se conectar
Exemplos	login morgana

Comando	logout e exit
Comentário	Utilizados para se desconectar ao sistema
Sintaxe	logout exit

Comando	halt
Comentário	Utilizado para desligar o computador
Sintaxe	halt

Comando	Reboot
Comentário	Utilizado para reiniciar o computador
Sintaxe	reboot

Comando	shutdown
Comentário	Utilizado para desligar ou reiniciar o computador
Sintaxe	shutdown [-r] [-h] now
Descrição	-r : para reiniciar o computador -h : para desligar o computador
Exemplos	shutdown -r now shutdown -h now

6.2) Gerenciamento de Arquivos e Diretório:

6.2.1) Listando Arquivos:

Comando	ls
Comentário	Utilizado para listar o conteúdo de um determinado diretório
Sintaxe	ls [parâmetros] [diretório] diretório: é o diretório a ser listado parâmetros: -l : Listagem completa, formato longo -a: lista os arquivos ocultos -h: exibe o tamanho dos arquivos em kbytes ou Mbytes
Exemplos	ls -lah /home/

6.2.2) Metacaracteres:

Os metacaracteres são geralmente utilizados na abreviação de nomes de arquivos, e seu uso é muito difundido no mundo Linux. Os principais metacaracteres são: ? * [] ;

Para que possamos entender melhor o seu funcionamento vamos supor que dentro do nosso diretório pessoal existam os seguintes arquivos:

```
teste1 teste2 teste3 teste4
texto01 texto02 texto03 texto04
```

Agora vamos ver como cada um trabalha:

“ ? ” - Pode ser utilizado para substituir qualquer caracter

Por exemplo: **ls teste?**

Obteríamos como resposta:

```
teste1 teste2 teste3 teste4
```

“ * ” - Pode ser utilizado para substituir qualquer seqüência de caracteres

Por exemplo: `ls te*`

Obteríamos como resposta:

```
teste1 teste2 teste3 teste4
texto01 texto02 texto03 texto04
```

“ [ab] ” - Define um conjunto de caracteres que poderão ser utilizados no processo de substituição.

Por exemplo: `ls teste[14]`

E obteríamos como resposta:

```
teste1 teste4
```

“ [a-b] ” - Define uma faixa de caracteres que poderão ser utilizados no processo de substituição.

Por exemplo: `ls teste[1-4]`

E obteríamos como resposta:

```
teste1 teste2 teste3 teste4
```

“ ; ” - Separa vários comandos em uma única linha

Por exemplo: `cd ~ ; ls -lah`

6.2.3) Criando, Removendo e Imprimindo Diretórios

Comando	<code>mkdir</code>
Comentário	Utilizado para criar diretórios
Sintaxe	<code>mkdir [nome_do_diretório]</code>
Exemplos	<code>mkdir ~/teste1 ~/teste2</code>

PS: é possível criar mais do que um diretório ao mesmo tempo.

Comando	<code>cd</code>
Comentário	Utilizado para alterar o diretório corrente
Sintaxe	<code>cd [path_para_o_diretório]</code>
Exemplos	<code>cd /etc/X11/</code>

Comando	<code>rmdir</code>
Comentário	Utilizado para remover um dado diretório
Sintaxe	<code>rmdir [path_para_o_diretório]</code>
Exemplos	<code>rmdir /etc/X11/</code>

OBS: para remover um diretório o mesmo deve estar vazio

Comando	pwd
Comentário	Utilizado para informar o diretório corrente
Sintaxe	pwd

6.2.4) Copiando, Removendo, Movendo e Renomeando Arquivos;

Comando	cp
Comentário	Utilizado para copiar arquivos
Sintaxe	cp -Rv [origem] [destino] - R : cópia recursiva, ou seja, se houver subdiretórios o conteúdo dos mesmos também serão copiados - v : exibe o que está sendo copiado no momento - f : modo forçado, sobre escreve o arquivo sem solicitar confirmação
Exemplos	cp /bin/bash ./

Comando	mv
Comentário	Utilizado para mover arquivos
Sintaxe	mv -Rv [origem] [destino] - R : move recursivamente - v : exibe o que está sendo movimentado - f : modo forçado, sobre escreve o arquivo sem solicitar confirmação
Exemplos	mv /home/aluno /home/aluno/bkp

Comando	rm
Comentário	Utilizado para remover arquivos
Sintaxe	rm -Rv [origem] - R : remove recursivamente - v : exibe o que está sendo apagado no momento - f : modo forçado, apaga o arquivo sem solicitar confirmação
Exemplos	rm /home/aluno /home/aluno/bkp

6.2.5) Exibindo o conteúdo de arquivos

Comando	cat
Comentário	Utilizado para exibir ou concatenar dois arquivos.
Sintaxe	cat [arquivos]
Descrição	[arquivos] é a lista de nome dos arquivos a serem exibidos ou concatenados
Exemplos	cat /etc/hosts cat lista1.txt lista2.txt >> lista-completa.txt

Comando	more
Comentário	Exibe o conteúdo de um arquivo, uma tela por vez.
Sintaxe	more [arquivo]
Descrição	Pressione “ENTER” para movimentar-se pelo arquivo
Exemplos	more /etc/passwd

Comando	less
Comentário	Permite que a exibição de um arquivo possa ser feita uma página por vez, e ainda a movimentação para cima e para baixo dentro do mesmo.
Sintaxe	less [arquivo]
Descrição	Pressione “ENTER” ou use as setas direcionais do teclado para movimentar-se pelo arquivo. Para sair deste comando pressione “q”
Exemplos\	less /etc/fstab

6.2.6) Links

Comando	ln
Comentário	cria links, ou seja, permite que vários atalhos apontem a um mesmo arquivo real, ou posição da árvore de diretórios. Existem basicamente dois tipo de links, são eles: <ol style="list-style-type: none"> 1.) Hard Links >> dois nomes para o mesmo i-node 2.) Link Simbólico >> arquivo especial que contem o caminho completo para o arquivo destino
Sintaxe	ln -s <origem> <link> <p>-s = cria link simbólico</p> <p>origem = arquivo ou diretório a se referido</p> <p>link = nome a ser dado para o arquivo de link</p>
Exemplos	cd~ cp /bin/bash ./ ln bash hteste ln -s bash steste ls -la

6.2.7) Comparando Arquivos

Comando	cmp
Comentário	utilizado para comparar o conteúdo de dois arquivos quaisquer.
Sintaxe	cmp [arquivo1] [arquivo2]
Descrição	este comando somente exibe a primeira diferença entre os dois arquivos
Exemplos	cat arq1.txt 1234567890 cat arq2.txt 1234567890X cmp arq1 arq2 arq1 arq2 differ: char 11, line 1

Comando	diff
Comentário	exibe a diferença entre dois ou mais arquivos
Sintaxe	diff -c [arquivo1] [arquivo2] diff -c [diretório1] [diretório2]
Exemplos	cat arq1.txt 123456789 1234567890 cat arq2.txt 1234567890 123456789 diff arq1 arq2 *** teste1 2003-02-03 23:21:20.000000000 -0200 --- teste2 2003-02-03 23:21:31.000000000 -0200 ***** *** 1,2 **** - 123456789 1234567890 --- 1,2 ---- 1234567890 + 123456789

6.3) Pedindo ajuda:

Comando	man
Comentário	Exibe a página de manual, help, de um determinado comando do Linux
Sintaxe	man [comando]
Descrição	[comando] é o nome do comando
Exemplos	man passwd

Comando	help
Comentário	Exibe informações sobre comandos internos do shell
Sintaxe	help [comando]
Descrição	[comando] é nome do comando interno
Exemplos	help help help echo

Comando	info
Comentário	Exibe informações sobre vários comandos, baseado na base de dados "info", este comando é muito semelhante ao man, mas este é orientado a menus.
Sintaxe	info [comando]
Descrição	[comando] é o nome do comando procurado
Exemplos	info info bash

6.4) Permissão dos Arquivos:

Comando	chmod
Comentário	modifica as permissões de um determinado arquivo ou diretório.
Sintaxe	chmod [permissões] [arquivo]
Descrição	[permissões] : direitos a serem atribuídos a um determinado arquivo Tipos de permissão: Classes: u = usuário g = grupo o = outros Tipo de permissões: r – leitura w – escrita x – executar Alteradores: + - adiciona - - remove = - iguala
Exemplos	cp /bin/.bash ~/teste ls -la chmod g+w teste ls -la chmod g-w teste ls -la
Comando	umask
Comentário	Utilizado para a definição da mascara padrão de criação de arquivos
Sintaxe	umask [mascara]
Descrição	[mascara] é o valor absoluto da mascara em octal
Exemplos	umask umask 0027 umask 0077
Comando	chown
Comentário	altera o proprietário, dono, de um ou vários arquivos
Sintaxe	chown [-fhR] [proprietário] [arquivo]
Descrição	-f: não reporta erros -h: se o arquivo for um link simbólico, altera a propriedade do link simbólico e não do arquivo. -R: opção recursiva
Exemplos	chown -R Irodrigo /home/Irodrigo/copia/uther/*

Comando	chgrp
Comentário	altera o grupo do usuário
Sintaxe	chgrp [-fhR] [gid] [arquivo]
Descrição	gid : é o número decimal que identifica o grupo, ou o nome do grupo -f : não reporta erros -h : se o arquivo for um link, altera o link e não o arquivo original -R : opção recursiva
Exemplos	chgrp -R Irodrigo /home/Irodrigo/copia/uther/*

6.5) Pipe e Redirecionadores

“ > ” - Envia a saída padrão do comando para um determinado arquivo

Por exemplo:

```
ls -lah
ls -lah > exemplo.txt
cat exemplo.txt
```

OBS: Se o já existir ele será sobre escrito pelo novo arquivo

“ >> ” - Envia a saída padrão do comando para um determinado arquivo, sem destruir o conteúdo original.

```
ls -lah > exemplo2.txt
cat exemplo.txt
ls -lah >> exemplo2.txt
cat exemplo2.txt
```

“&>” Redireciona, ao mesmo tempo, a saída padrão e a saída de erro padrão. Este é muito utilizado quando estamos compilando um programa.

```
gcc -o teste teste.c &> mensagens.txt
```

“ 2> ” Redireciona somente a saída de erro padrão.

```
gcc -o teste teste.c 2> erro.txt
```

OBS: É possível gerar dois arquivos independentes, ou seja saída padrão em um e saída de erro padrão em outro.

Por exemplo:

```
gcc -c teste teste.c 2> erro.txt > /dev/null
```

“ < ” - Faz com que os comandos obtenham a sua entrada a partir de um arquivo, mas raramente é utilizado

```
sort < origem.txt > saida.txt
```

OBS: O comando acima organiza o conteúdo do arquivo “origem.txt” e envia a sua saída para o arquivo “saida.txt”

“ | ” - faz com que um dado comando opere com a saída de um outro, redireciona a saída de um comando para a entrada de outro.

du | sort -n

du | sort -rn

du | sortt -rn | more

6.6) Editores de Texto

6.6.1) Editor de Texto VI

Este é um dos editores de texto mais usados no mundo linux (Unix), ele é composto basicamente em dois modos de operação, são eles:

1. **Modo texto:** no qual é realizada a “inserção” do texto, para sair deste modo basta teclar “Esc”;
2. **Modo Comando:** utilizado para inserção dos comandos, é neste modo que o programa está quando carregado;

A seguir tem-se uma listagem dos principais comandos destes modos de operação

6.6.1.1) Comandos de Inserção de texto

i : insere o texto no início da linha

A: insere texto no final da linha

o: adiciona uma linha abaixo da corrente

O: adiciona uma linha acima da corrente

6.6.1.2) Comandos de Movimentação de texto – Modo Comando

ctrl+f : pula para a tela seguinte

ctrl+b: pula para a tela anterior

H : primeira linha da tela

L : última linha da tela

^ : move para o 1º caracter não branco da linha

\$: final da linha corrente

nG: move para a linha “n”

G: move para a última linha do arquivo

k: uma posição acima

j: uma posição abaixo

h: uma posição a esquerda

l: uma posição a direita

Comando	updatedb
Comentário	Atualiza a base de dados utilizada pelo comando slocate, para localizar arquivos
Sintaxe	updatedb -u
Descrição	-u : inicia a atualização a partir do diretório "/"
Exemplos	Updatadb -u

Comando	slocate
Comentário	Utilizado para localizar arquivos no computadores, baseado em na listagem de arquivos armazenada em uma base de dados
Sintaxe	slocate -u -n [numero] [string]
Descrição	[string] é parte do nome do arquivo que se deseja localizar -u : é utilizado para atualizar a base de dados -n [numero] : especifica um número máximo de resultados
Exemplos	slocate -n5 find

Comando	which
Comentário	Procura programas executáveis ou shell scripts que estão em um dos diretórios do caminho de busca, ou seja, os diretórios que estão presentes na variável PATH
Sintaxe	which [comando]
Descrição	[comando] é o nome do comando ou shell script que se deseja localizar
Exemplos	which bash which passwd

Comando	whereis
Comentário	tenta localizar o arquivo binário, o código fonte e as páginas de manual de um ou mais programas
Sintaxe	whereis [-bms] [programa]
Descrição	-b = apenas o arquivo binário -m = apenas as páginas de manual -s = apenas o código fonte
Exemplos	whereis -bms reset

6.8) Manipulação de Texto

Comando	sort
Comentários	ordena um conjunto de dados fornecidos e os retorna para a saída padrão.
Sintaxe	sort <arquivo>
Exemplos	sort /etc/passwd

Comando	wc
Comentário	Realiza uma contagem de caracteres, palavras e linhas de um determinado arquivo
Sintaxe	wc -c -l -w [arquivo]
Descrição	-c: caracteres; -l: linhas; -w: palavras.
Exemplos	wc -cwl /etc/passwd

Comando	head
Comentário	Imprime as primeiras linhas de um arquivo.
Sintaxe	head -n num [arquivo]
Descrição	-n: número de linhas (num)
Exemplos	head -n 15 /etc/passwd

Comando	tail
Comentário	Imprime as últimas linhas de um arquivo.
Sintaxe	tail -n [arquivo]
Descrição	-n: número de linhas (num)
Exemplos	tail /etc/passwd

Comando	cut
Comentário	mostra colunas específicas do texto
Sintaxe	cut -f -d <arquivo>
Descrição	-f = colunas a serem exibidas -d = delimitador de colunas utilizado
Exemplos	cut -f=1,3 -d=: arquivo.txt

Comando	grep
Comentários	dado um grupo de arquivos, imprime as linhas destes arquivos que seguem um determinado padrão
Sintaxe	grep [padrão] [arq1[arq2 [arq3 [... [arqn]]]]]
Exemplos	grep root /etc/passwd

Comando	expand
Comentário	Converte os "tab" do arquivo para espaço em branco
Sintaxe	expand [arquivo]
Descrição	[arquivo] é o nome do arquivo a ser modificado
Exemplos	expand teste.txt

6.9) Sistema de Arquivo

Comando	fdisk
Comentário	Este comando é utilizado no gerenciamento das partições de disco
Sintaxe	fdisk -lus [particao]
Descrição	[partição] é a descrição da partição de disco que se deseja examinar -l : lista todas as partições de disco existentes na máquina -u : exhibe o tamanho em setores em vez de cilindros -s : exhibe o tamanho em blocos de partição
Exemplos	fdisk -l fdisk -lu fdisk -s /dev/hda2

Comando	mkfs
Comentário	Este comando é utilizado para criar os sistemas de arquivo das partições de disco
Sintaxe	mkfs --t fstype -c filesystem
Descrição	-f [fstype]: especifica o tipo de sistema de arquivo a ser criado -c : realiza a verificação de blocos ruins antes de criar o sistema de arquivo filesystem : é a partição a ser manipulada
Exemplos	Mkfs -t ext3 -c /dev/hda10

Comando	fsck
Comentário	Verifica e recupera o sistema de arquivo no linux
Sintaxe	fsck -Aa [filesystem]
Descrição	[filesystem] : é o sistema de arquivo, ou partição, a ser examinada -A : verifica todos os sistemas de arquivo existentes no arquivo "/etc/fstab" -a : repara o sistema de arquivo sem fazer nenhuma pergunta
Exemplos	fsck -a /dev/hda2

Comando	du
Comentário	exibe o consumo de espaço em disco de um determinado diretório ou arquivos
Sintaxe	du -hks [objeto]
Descrição	-h : exhibe o tamanho em formato humano -k : exhibe o tamanho em kbytes -h : realiza a soma do espaço em disco de um determinado segmento, ou diretório
Exemplos	du -hs ~/*

Comando	df
Comentário	Exibe a quantidade de espaço livre em disco, em uma determinada partição ou em todas as partições montadas
Sintaxe	df -hk [ponto_de_montagem]
Descrição	-h : exibe as informações no formato humano -k : exibe as informações em kbytes [ponto_de_montagem] é o diretório que está associado a um sistema de arquivo
Exemplos	df -h df -k df -h /home

Comando	mount
Comentário	Anexa um determinado sistema de arquivo em um diretório especificado.
Sintaxe	<i>mount [opções] [dispositivo_especial] [diretório]</i>
Descrição	opções: - a : monta todos os sistemas de arquivo que estão declarados no arquivo /etc/fstab; - r : monta o sistema de arquivo no modo read-only; - w : monta o sistema de arquivo no modo read-write; -t <i>file_system</i> : indica o sistema de arquivo a ser montado, os principais sistemas de arquivo são: vfat : utilizado pelo Windows 9x e DOS; ext2: utilizado pelas versões antigas do Linux; ext3: utilizado pelas versões novas do Linux; nfs: sistema de arquivo distribuído pela rede; iso9660: sistema de arquivo padrão dos CD-R e CD-RW ntfs: utilizado pelo Windows NT, XP e 2000 [dispositivo_especial] /dev/hdxn x: pode assumir os valores a,b,c,d; n: de 1 até 4 para partições primárias e > 4 para partições estendias; [diretório] : é o diretório no qual o dispositivo deve ser montado
Exemplos	mount -a mount -t ntfs /dev/hda1 /mnt/win_c mount -t vfat/dev/hda2 /mnt/win_d mount /mnt/cdrom mount /mnt/floppy mount -t nfs tux-pet:/hom /home

Comando	umount
Comentário	Desmonta um dado sistema de arquivo que está anexado ao diretório especificado.
Sintaxe	umount [opções] [diretório]
Descrição	opções: -a : desmonta todos os sistemas de arquivo que estão declarados no arquivo etc/fstab; -t <i>file_system</i> : indica o sistema de arquivo a ser desmontado; <i>[diretório]</i> : é o diretório no qual o sistema de arquivo esta montado ou o próprio sistema de arquivo.
Exemplos	umount -a umount /mnt/win_c umount /mnt/win_d umount /mnt/cdrom umount /mnt/floppy

6.10) Gerenciamento de Processos

Comando	ps
Comentário	Este comando é utilizado para exibir “todos” os processos que estão atualmente rodando na máquina
Sintaxe	ps [opções]
Descrição	Opções: -a : os processos que pertencem ao usuário -u : informações no modo de saída amplo -x : processos sem um terminal de controle
Exemplos	ps -aux

Comando	kill
Comentário	Envia um determinado sinal para um processo, geralmente utilizamos o sinal -9 que serve para matar o processo que é o alvo do sinal.
Sintaxe	kill [opção] [pid]
Descrição	<i>pid</i> : é o id do processo, opções: -l : lista os nomes dos sinais -sinal : número do sinal, os mais importantes são: 9 : Mata o processo; HUP: recarrega um processo
Exemplos	kill -hup `cat /var/run/mysqld/mysqld.pid`

Comando	killall
Comentário	Envia um determinado sinal para todos os processos que possuem o mesmo nome.
Sintaxe	kill [-ei] [-l] [-signal] [nome]
Descrição	nome : é o nome do processo, opções: -l : lista os nomes dos sinais -e: envia um sinal apenas para as combinações exatas de nome -i : solicita uma confirmação antes de enviar um sinal - <i>sinal</i> : número do sinal, os mais importantes são: HUP: Recarrega o processo. 9 : Mata o processo;
Exemplos	killall -HUP named

Comando	top
Comentários	Exibe os processos que mais estão utilizando CPU no momento.
Sintaxe	top

Comando	jobs
Comentário	Lista todos os trabalhos que estão sendo executados ou que estão suspensos/parados.
Sintaxe	jobs [-l]
Descrição	- l : inclui os ids dos processos
Exemplos	jobs -l

Comando	fg
Comentários	utilizado para trazer um comando que estava sendo executado no segundo plano para o primeiro plano
Sintaxe	fg
Exemplos	find / -name linuxconf -print & fg find

Comando	bg
Comentários	faz com que um comando seja executado em segundo plano, geralmente útil quando o programa irá tomar muito tempo para ser executado.
Sintaxe	bg
Exemplos	find / - name lilo.conf > local.txt CTRL+Z bg

Comando	&
Comentários	assim como o anterior, faz com que um comando seja executado em segundo plano, porém neste caso, ele não será executado no primeiro plano antes.
Sintaxe :	[comando] &
Exemplos	find / - name lilo.conf > local.txt &

Comando	control+c
Comentários	utilizado para matar um processo que está rodando no primeiro plano

Comando	control+z
Comentários	utilizado para parar um processo que está rodando no primeiro plano

6.11) Compactação e Backup

Comando	gzip
Comentários	este comando é utilizado para compactar arquivos, sendo que geralmente o arquivo compactado substitui o arquivo original
Sintaxe :	gzip [-cdfirt] [arquivo]
Descrição	-c = deixa o arquivo original intacto -d = descompacta o arquivo -f = força a compactação mesmo quando já existe um arquivo com o mesmo nome -l = lista os arquivos que estão dentro do arquivo compactado -r = compacta recursivamente descendo na estrutura de diretórios
Exemplos	cp /etc/passwd ~/senhas ls -l gzip senhas ls -l gunzip senha.gz ls -l gzip -c senhas > senhas_compactadas.gz ls -l gunzip senhas_compactadas.gz ls -l

Comando	gunzip
Comentários	descompacta um arquivo compactado pelo gzip, ou seja, aqueles arquivos que possuem extensão .gz
Sintaxe :	gunzip [-cdfirt] [arquivo]
Descrição	vide comando anterior
Exemplos	vide comando anterior

Comando	compress
Comentários	comando utilizado para compactar arquivos
Sintaxe :	compress [-vr] [arquivo]
Descrição	-v = apresenta a porcentagem de redução no tamanho do arquivo -r = opção recursiva, somente deve ser utilizada quando o arquivo especificado for um diretório
Exemplos	compress -rv ~/teste

Comando	uncompress
Comentários	descompacta arquivo compactados com o compress, ou seja aquelas que possuem extensão .Z
Sintaxe	uncompress [-rv] [arquivo]
Descrição	vide comando anterior
Exemplos	uncompress -rv ~/teste

Comando	bzip2
Comentários	Comando utilizado para realizar a compressão de arquivos, possui uma taxa de compressão superior ao gzip
Sintaxe	bzip2 -dtv19 [arquivo]
Descrição	[arquivo] é o nome do arquivo a ser manipulado -d : descomprime o arquivo -t: testa a integridade do arquivo -v: mostra a taxa de compressão obtida -9: melhor taxa de compressão -1: compressão rápida
Exemplos	cd ~ ; mkdir teste ; cd teste cp -v /bin/u* ./ ls -l bzip * ls -l bunzip *.bz2

Comando	bunzip2
Comentários	descomprime arquivos compactados pelo comando bzip2
Sintaxe	bunzip2 [arquivo]
Descrição	vide comando anterior
Exemplos	vide comando anterior

Comando	zip
Comentários	Compactador de arquivos “padrão” do Windows
Sintaxe	zip -FPRT [senha]
Descrição	-F: corrige o arquivo -P [senha] : acrescenta uma senha no arquivo -R: compactação recursiva -T: testa a integridade do arquivo
Exemplos	cd ~ ; mkdir teste; cd teste zip -P senha binários /bin/*

Comando	unzip
Comentários	descomprime um arquivo criado com o comando zip
Sintaxe	unzip [arquivo]
Descrição	[arquivo] é o nome do arquivo compactado
Exemplos	cd ~ ; mkdir teste; cd teste unzip binarios.zip

Comando	tar
Comentários	este comando cria um arquivo, pacote, contendo um conjunto de arquivos, mas este pacote não está obrigatoriamente compactado
Sintaxe	tar [-cxzvf] [destino] [origem]
Descrição	-c = cria o arquivo -x = extrai o arquivo -z = compacta o arquivo -v = modo “falador” ou verbose -f = especifica o nome do pacote a ser criado
Exemplos	cd ~ ; mkdir teste cp -v /bin/* ~/teste tar -cvf pacote.tar ~/teste/ ls -l ; cd ~/teste rm * cd .. rmdir ~/teste tar -xvf pacote.tar cd ~/teste ls -l cd ~ tar -xzvf pacote2.tgz ~/teste/

6.12) Data e Hora

Comando	cal
Comentários	Utilizado para exibir o calendário de um determinado mês.
Sintaxe	cal -y [mês [ano]]
Descrição	-y: calendário de todo o ano mês: calendário do mês
Exemplos	cal 02 1994

Comando	date
Comentários	Utilizado para exibir e alterar a data e a hora atual
Sintaxe	date [MMDDhhmm[[CC]YY]
Descrição	MM : mês com dois dígitos DD: dia com dois dígitos hh: hora com dois dígitos mm: minuto com dois dígitos CCYY: ano com quatro dígitos
Exemplos	date 021516301994 date +%D date +%y-%m-%d date +%Y-%m-%d

6.14) Comandos de Vídeo

Comando	clear
Comentários	Utilizado para limpar a tela
Sintaxe	clear
Exemplos	clear

Comando	echo
Comentários	Utilizado para exibir mensagens na tela
Sintaxe	echo [valor]
Descrição	valor: pode ser um texto ou uma variável
Exemplos	echo \$PATH

7) Lista de Exercícios:

Comentários sobre a Lista de exercícios:

- (1) Ela pode ser feita em dupla;
- (2) A lista deve conter o nome e a turma dos componentes do grupo;
- (3) Somente alunos da mesma turma podem fazer parte de um grupo;
- (4) Ela só será aceita até a data determinada acima.

Lista – Gerenciando Arquivos I

- 1) Construa a seguinte estrutura de diretório:
- 2) Informe qual é a seqüência de comandos necessária para criar o arquivo **listagem.txt**, contendo a listagem de todos os arquivos do diretório atual, incluindo os arquivos ocultos, no formato longo.
- 3) Informe qual é o comando que copia para o diretório **/home/lab1/backup** todos os arquivos que estão do diretório **/etc**, bem como de todos os que estão em seus subdiretórios.
- 4) Informe qual é o comando que apaga todos os arquivos do diretório **/etc/lixo**, sem que nada seja questionado ao usuário e que ainda gere o arquivo **apagados.txt**, com a listagem de todos os arquivos que foram removidos.
- 5) Informe qual é a seqüência de comandos necessária para: armazenar no arquivo **calendario.txt** o calendário do mês de agosto de 1976.
- 6) Informe qual é o comando utilizado para exibir na tela o conteúdo do arquivo **/etc/inittab**, uma tela por vez.
- 7) Informe qual é o comando necessário para exibir na tela o conteúdo do arquivo **/etc/passwd**, sem que o mesmo seja paginado.

Lista - Gerenciamento de Arquivo II

- 1) Informe o comando que lista as 10 primeiras linhas do arquivo **/etc/passwd**.
- 2) Informe o comando que lista as 5 últimas linhas do arquivo **/var/log/message**.
- 3) Informe qual é a seqüência de comandos que lista da 3ª a 5ª linha do arquivo **/etc/inittab**.
- 4) Informe o comando necessário para se criar um link simbólico com o nome de teste.lnk para o arquivo **/bin/bash**.
- 5) Informe a seqüência de comandos necessária para exibir na tela da linha 10 até a linha 10 do arquivo **/var/log/messages**.

Lista – Gerenciamento de Arquivos III

- 1) Informe qual e a seqüência de comandos para: procurar por todos os arquivos que estão no computador do usuário e possuem o nome igual a “exemplo.txt”.
- 2) Informe qual e a seqüência de comandos para: procurar por todos os arquivos que estão dentro do “HomeDir” do usuário e que possuem como extensão “c” e em seguida redirecionar o nome dos arquivos encontrados para o arquivo “arquivos.txt” e as mensagens de erro para o arquivo “erros.txt” ambos localizados no “HomeDir” o usuário.
- 3) Informe qual é a seqüência de comandos para: armazenar no arquivo “comando.txt” a localização do arquivo binário, do código fonte e das paginas de manual do comando “slocate”.
- 4) Qual é o comando que cria o arquivo /home/lab1/log.txt, sem nenhum conteúdo e com a data e a hora atual.
- 5) Informe qual é a seqüência de comandos necessários para se criar o arquivo usuarios.txt, contendo a listagem, em ordem alfabética, de todos os usuários cadastrados na máquina.
- 6) Informe a seqüência de comandos necessários para armazenar no arquivo usuarios2.txt a primeira e a segunda coluna do arquivo /etc/passwd.
- 7) Informe a seqüência de comandos necessária para ordenar o arquivo /etc/passwd, na ordem alfabética inversa, e exibir na tela a 3ª e a 4ª linha.
- 8) Informe a seqüência de comandos necessária para criar o arquivo ./10primeiros.txt com as 10 primeiras linhas do arquivo /etc/passwd ordenado alfabeticamente.
- 9) Informe a seqüência de comandos utilizada para criar o arquivo ./lista-ordenada.txt com o conteúdo do arquivo /etc/group ordenado alfabeticamente. Em seguida copie a 1ª, 5ª, 2ª, 9ª e 4ª linhas, deste arquivo que acabou de ser criado e nesta ordem, para o arquivo ./nova-lista.txt.
- 10) Informe a seqüência de comandos necessária para:
 - Entrar no HomeDir do usuário;
 - Criar um diretório chamado copia;
 - Entrar no diretório copia;
 - Copiar para o diretório atual de forma recursiva todos os arquivos que estão no diretório /var/log para o diretório atual;
 - Mudar o proprietário de todos os arquivos do diretório atual, bem como de seus subdiretórios, para a usuária “ssattler”;
 - Mudar o grupo de todos os arquivos do diretório atual, bem como de seus subdiretórios, para o grupo “professores”;
 - Mudar a permissão de todos os arquivos do diretório atual, bem como de seus subdiretórios, de tal forma que o proprietário tenha acesso total, o grupo tenha acesso de leitura e as demais pessoas não tenham nenhum tipo de acesso;

- 1) Qual é o comando que lista na tela todos os processos, com e sem terminal no formato amplo, que pertencem ao usuário ROOT?
- 2) Qual é o comando que lista na tela todos os processos, cujo nome é Bash no formato amplo, ordenados pelo nome do usuário e não pelo PID?
- 3) Qual é o comando que lista na tela todos os processos, com e sem terminal, no formato amplo uma tela por vez?
- 4) Qual é a linha de comando que envia para o arquivo "~/processos.txt" todos os processos, com e sem terminal, no formato amplo e ordenados pelo nome do usuário que o disparou?
- 5) Quais são os comandos necessários para se descobrir o PID, somente dos processos cujo nome é Bash, e então enviar um sinal para matar todos os processos Bash que pertencem ao usuário LAB1?
- 6) Qual é o comando que envia um sinal para interromper, parar a execução, de todos os processos cujo nome é wget?
- 7) Qual é o comando que acorda, volta a fazer o processo rodar, todos os processos cujo nome é wget?
- 8) Qual é o comando que gera um processo que irá rodar somente no segundo plano e no final de sua execução terá gerado o arquivo "~/mp3.txt" contendo uma listagem de todos os arquivos ".mp3" que existem na máquina?
- 9) Qual é a seqüência de passos necessários para executar o comando "wget <http://www.dicas-unicamp.br/>" inicialmente no primeiro plano, e em seguida enviar este processo para o segundo plano?
- 10) Qual é o comando que gera o arquivo ~/trabalhos_na_fila.txt contendo os dados de todos os processos que estão na fila de execução ou na fila de espera.

- 01) Qual é o comando necessário para criar um pacote “tar”, com o nome de “backup.tar” e contendo todos os arquivos do homedir do usuário. A saída deste comando deve ser redirecionada para o arquivo ~/arquivos.txt?
- 02) Qual é o comando necessário para criar um pacote “tar”, com o nome de “backup.tar”, contendo todos os arquivos do homedir do usuário. A saída deste comando deve ser redirecionada para o arquivo ~/arquivos.txt de tal forma que todos os arquivos fiquem na ordem alfabética.
- 03) Quais são os comandos necessários para realizar o backup do diretório /etc para o pacote “~/backup.tar”, de tal forma que a sua saída seja direcionada para o arquivo “~/lista_arquivo.txt”. Em seguida, devemos incluir os arquivos do diretório /bin no pacote “~/backup.tar” e a saída deste segundo comando também deve ser direcionado para o arquivo “~/lista_arquivo.txt”?
- 04) Qual é o comando necessário para gerar o pacote “backup.tgz” contendo todos os arquivos do diretório “/usr/local/bin”?
- 05) Utilizando-se o gunzip e o tar, como proceder para conseguir acessar os arquivos que estão dentro do pacote “backup.tar.gz”
- 06) Qual é o comando necessário para gerar o pacote ~/bmp.tar.Z contendo todos os arquivos .bmp do diretório atual?
- 07) Qual é o comando utilizado para localizar todos os arquivos que pertencem ao usuário “Lab1”, que estão dentro do diretório /home e em seguida gerar um pacote “.tar” contendo todos estes arquivos
- 08) Qual é o comando utilizado para localizar todos os arquivos que pertencem ao grupo “alunos”, que estão dentro do diretório /home e em seguida gerar um pacote “.tar” contendo todos estes arquivos. Feito isto compacte este arquivo com o comando gzip?
- 9) Qual é o comando utilizado para localizar todos os arquivos do tipo MP3, que estão dentro do diretório /home e em seguida gerar um pacote “.tar” contendo todos estes arquivos. Finalmente compacte este arquivo dando origem a um arquivo com terminação “.Z”
- 10) Qual é o comando utilizado para localizar todos os arquivos que não pertencem a nenhum usuário válido, que estão dentro do diretório /home e em seguida gerar um pacote “.tar” contendo todos estes arquivos. Finalmente compacte este arquivo com o comando Bzip?
- 11) Informe qual é a seqüência de comandos necessários para:
 - Entrar no HomeDir do usuário;
 - Criar um diretório chamado copia;
 - Entrar no diretório copia;
 - Copiar para o diretório atual de forma recursiva todos os arquivos que estão no diretório /usr/local para o diretório atual;
 - criar um pacote com o nome de “programas.tar” contendo todos os arquivos do diretório atual, bem como dos seus subdiretórios;
 - E em seguida compactar este arquivo como o bzip2
- 12) Informe qual é a seqüência de comandos necessários para:
 - Entrar no HomeDir do usuário;
 - Criar um diretório chamado “/restore”;
 - Descompactar o arquivo “/backup/programas.tar.gz”
 - Entrar no diretório “restore” e extrair os arquivos que estão dentro do pacote que foi descompactado pelo comando anterior

- 1) Qual é a vantagem do Unix sobre os sistemas operacionais anteriores a ele?
- 2) Em qual linguagem foi desenvolvida a primeira geração do Unix?
- 3) Qual foi a vantagem de se utilizar a linguagem “C” na segunda geração do Unix?
- 4) O processo de instalação de um Linux pode ser dividido em quais etapas?
- 5) Quais são as informações necessárias para se ter acesso a uma determinada partição de disco? Mostre um exemplo de como se realiza esta identificação.
- 6) Quais são os comandos associados a “diretórios” ? Qual é a função de cada um?
- 7) Quais são todos os comandos associados a copia, remoção e movimentação de arquivos? Dê um exemplo do funcionamento de cada um e explique o que ele está fazendo.
- 8) Quais são todos os comandos que podemos utilizar para visualizar um determinado arquivo? Dê um exemplo de cada um.
- 9) Quais são todos os redirecionadores de entrada e saída? Qual é a função de cada um? Dê um exemplo para uso de cada.
- 10) Quais são todos os metacaracteres? Qual é a função de cada um? Dê um exemplo do uso para cada.