



**INSTITUTO FEDERAL**  
**SANTA CATARINA**



# **ICO60801**

## **Introdução à Computação**

***Aula 08***

***Linux em modo texto***

**Professores:** Maria Claudia Castro / Bruno Fontana  
**Semestre letivo:** 2014-1



# Vantagens

- Simplicidade
  - Configurações através de arquivo de configuração
- Estabilidade
- Baixo requerimento de recursos
- Automatização de processos com scripts



# O Shell

- **Programa** interpretador de comandos do Linux
- Analisa o texto digitado e produz um resultado
  - Texto é *case sensitive*
    - “cd”, “cD”, “Cd” e “CD” são coisas diferentes
- Exemplos de interpretadores: ksh, csh, bash, etc



# Extensão de arquivos

- No Windows:
  - arquivo.txt → Bloco de notas
  - arquivo.bmp → Paint
  - arquivo.doc → Word
- No Linux, arquivos não precisam ter extensão
  - Informação do tipo do arquivo fica no cabeçalho
  - Extensões servem para informação ao usuário
  - Alguns ambientes gráficos ainda utilizam extensões





# Extensão de arquivos: exemplo - comando *file*

```
user@maquina:~$ ls -l
total 856
-rw-r--r-- 1 user user 213072 2011-04-23 21:07 Figura1
-rw-r--r-- 1 user user 213072 2011-04-23 21:07 Figura2.png
-rw-r--r-- 1 user user 213072 2011-04-23 21:07 Figura3.jpg
-rw-r--r-- 1 user user 213072 2011-04-23 21:07 Figura4.txt
-rw-r--r-- 1 user user   302 2011-04-23 22:28 Texto1
-rw-r--r-- 1 user user   302 2011-04-23 22:28 Texto2.mp3
```

```
user@maquina:~$ file *
Figura1:      PNG image, 950x632, 8-bit/color RGB
Figura2.png:  PNG image, 950 x 632, 8-bit/color RGB
Figura3.jpg:  PNG image, 950 x 632, 8-bit/color RGB
Figura4.txt:  PNG image, 950 x 632, 8-bit/color RGB
Texto1:       ASCII text
Texto2:       ASCII text
```



# Arquivos ocultos

- Arquivo oculto começa com o carácter “.”  
Exemplo .bashrc, .cache, .bash\_history, etc

```
user@maquina:~$ ls -l
total 216
-rw-r--r-- 1 user user 213072 2011-04-23 23:46 Figura1
-rw-r--r-- 1 user user  302 2011-04-23 23:46 Texto1
```

```
user@maquina:~$ ls -la
total 864
drwxr-xr-x 2 user user 4096 2011-04-23 23:48 .
drwxr-xr-x 43 user user 4096 2011-04-23 23:46 ..
-rw-r--r-- 1 user user 213072 2011-04-23 23:46 Figura1
-rw-r--r-- 1 user user 213072 2011-04-23 23:46 .Figura2.png
-rw-r--r-- 1 user user 213072 2011-04-23 23:46 .Figura3.jpg
-rw-r--r-- 1 user user 213072 2011-04-23 23:46 .Figura4.txt
-rw-r--r-- 1 user user  302 2011-04-23 23:46 Texto1
-rw-r--r-- 1 user user  302 2011-04-23 23:46 .Texto2
```



# Tab completion

- Recurso que completa o texto digitado após um TAB
- O shell vê as opções disponíveis e completa o texto

```
user@maquina:~$ ls
```

```
Pasta_com_nome_grande_e_dificil_de_digitar Musicas
```

```
user@maquina:~$ cd Pas (TAB)
```

```
user@maquina:~$ cd Pasta_com_nome_grande_e_dificil_de_digitar
```

- Tendo mais de um arquivo com nome parecido, o shell completa até o ponto onde os nomes ficam diferentes
- Outro TAB mostra todas as opções

- Manual do sistema, geralmente em inglês
- Geralmente, todas as dúvidas sobre um comando podem ser sanadas com a leitura do *man*
- Sintaxe: *man <comando>*
- Controle:
  - Navegação pelo manual           setas acima e abaixo
  - Sair do manual                   q
  - Primeira linha                   g
  - Última linha                   G
  - Procurar texto                   /texto





- **NAME** – nome do comando
  - Exp: ls - list directory contents
- **SYNOPSIS** – expressão que resume todo o comando pesquisado
  - ls [OPTION]... [FILE]...
- **DESCRIPTION** – descrição detalhada do comando e da sua sinopse
  - -a, --all do not ignore entries starting with .
  - -A, --almost-all do not list implied . and ..
  - --author with -l, print the author of each file
- **NOTES** – alguns pontos importantes no comando
- **AUTHOR** – quem desenvolveu o programa
- **SEE ALSO** – comandos relacionados e mais informações

- Comando parecido com o man
  - *whatis* comando Imprime uma descrição curta do comando



# O superusuário

- Usuário administrador do sistema
- Conhecido como usuário *Root*
- Acesso a todos os arquivos e configurações do sistema
- **CUIDADO! Com o usuário root você pode danificar irreversivelmente o sistema com apenas alguns comandos**
- Login: root                      Senha: root
- Comando para se logar como root: *su -*



- Em versões recentes do Ubuntu
  - Usuário root desabilitado por padrão
  - Para habilitar: *sudo passwd*
- O comando *sudo*:
  - Forma de executar um comando com um usuário diferente do root

```
user@maquina:~$ ls /root
```

```
ls: não foi possível abrir o diretório /root: Permissão negada
```

```
user@maquina:~$ sudo ls /root
```

```
Área de Trabalho Desktop      Downloads  ModelosPúblico  
Compartilhado      Documentos Imagens      Música Vídeos
```



# Comandos - LS - Forma completa

```
user@maquina:~$ ls -l
```

```
total 4
```

```
-rwxr-xr-x 1 user user 179 2011-04-23 17:53 Arquivo.txt
```

1	2	3	4	5	6	7	8
-	rwxr-xr-x	1	user	user	179	2011-04-23 17:53	Arquivo.txt

Saída:

- **Total 4** - total em kBytes da pasta
- Campos
  - 1 “-” para arquivo e “d” para diretório
  - 2 esquema de permissão do arquivo
  - 3 número de diretórios dentro do arquivo
  - 4 e 5 usuário e grupo donos do arquivo
  - 6 tamanho do arquivo em blocos de 1kBytes
  - 7 data de modificação do arquivo
  - 8 nome do arquivo





# Comandos – CP (copy)

*cp* [*opcoes*][origem][destino] - copia origem para destino

- Opções interessantes:

- -r copia diretórios dentro de diretórios recursivamente

```
user@maquina:~$ ls
```

```
arquivo1.txt Pasta
```

```
user@maquina:~$ cp arquivo1.txt arquivo2.txt
```

```
user@maquina:~$ ls
```

```
arquivo1.txt arquivo2.txt Pasta
```

```
user@maquina:~$ cp arquivo1.txt Pasta/arquivo3.txt
```

```
user@maquina:~$ ls Pasta
```

```
arquivo3.txt
```



# Comandos – MV (move)

***mv** [opcoes][origem][destino] - move origem para destino*

- Pode ser usado também para renomear arquivos

```
user@maquina:~$ ls
```

```
arquivo.txt      Pasta
```

```
user@maquina:~$ mv arquivo.txt Pasta
```

```
user@maquina:~$ ls
```

```
Pasta
```

```
user@maquina:~$ cd Pasta
```

```
user@maquina:~/Pasta$ ls
```

```
arquivo.txt
```

```
user@maquina:~/Pasta$ mv arquivo.txt outro_nome.txt
```

```
user@maquina:~/Pasta$ ls
```

```
outro_nome.txt
```



# Diretórios especiais

- Diretório Raiz (/)
  - Diretório principal do sistema, onde estão localizados todos os outros diretórios

- **Comando:**
  - **aluno@maquina:~\$ cd /**
  - **aluno@maquina:/\$ ls -l**



# Diretórios especiais

- Diretório Atual (.)
  - Diretório onde nos encontramos no momento
- **Comando:**
  - **aluno@maquina:~\$ cd /home/aluno**
  - **aluno@maquina:~\$ pwd**
  - **aluno@maquina:~\$ cd .**
  - **aluno@maquina:~\$ pwd**





# Diretórios especiais

- Diretório Home (~)
  - Diretório pessoal do usuário
  - Caminho: /home/<login>

- **Comando:**
  - **aluno@maquina:~\$ cd /**
  - **aluno@maquina:~\$ pwd**
  - **aluno@maquina:~\$ cd ~**
  - **aluno@maquina:~\$ pwd**



# Diretórios especiais

- Diretório superior (..)
  - Diretório acima do diretório atual na árvore

## • Comando:

- `aluno@maquina:~$ cd /home/aluno`
- `aluno@maquina:~$ pwd`
- `aluno@maquina:~$ cd ..`
- `aluno@maquina:~$ pwd`



# Diretórios especiais

- Diretório anterior (-)
  - Último diretório utilizado

## • Comando:

- `aluno@maquina:~$ cd /home/aluno`
- `aluno@maquina:~$ pwd`
- `aluno@maquina:~$ cd /etc`
- `aluno@maquina:~$ pwd`
- `aluno@maquina:~$ cd -`
- `aluno@maquina:~$ pwd`



# Caminho **completo** e **relativo**

• Formas diferentes de se navegar em diretórios

• **Comando:**

- **aluno@maquina:~\$ cd /home/aluno**
- **aluno@maquina:~\$ pwd**
- **aluno@maquina:~\$ cd /**
- **aluno@maquina:~\$ pwd**
- **aluno@maquina:~\$ cd home**
- **aluno@maquina:~\$ pwd**
- **aluno@maquina:~\$ cd aluno**
- **aluno@maquina:~\$ pwd**