

### Propostas de atividades

Curso Técnico Integrado em Telecomunicações  
DISCIPLINA: **Introdução a Computação (ICO)**  
PROFESSORES: Bruno Fontana e Maria Cláudia de Almeida Castro  
CONTATO: [bruno.fontana@ifsc.edu.br](mailto:bruno.fontana@ifsc.edu.br) / [claudiacaastro@ifsc.edu.br](mailto:claudiacaastro@ifsc.edu.br)  
SEMESTRE: **2014 - 2**  
ENCONTROS: **Segunda-feira e Sexta-feira (07h30min)**

Nome do aluno: \_\_\_\_\_

## Comandos linux para manipulação de diretórios

### **ls - (list) Lista diretórios e seus conteúdos.**

\$ ls -a Lista os diretórios, arquivos ocultos e executáveis.

\$ ls -l Lista o conteúdo do diretório corrente detalhadamente , cada arquivo em uma linha, mostrando os atributos dos arquivos listados .

### **cd – (change directory) Mudando o diretório de trabalho.**

\$ cd ~ retornará ao diretório de entrada (diretório home do usuário)

\$ cd / retornará ao diretório raiz.

\$ cd - retornará ao diretório anteriormente acessado.

\$ cd .. sobe um diretório.

### **pwd – (present working directory) Identificação do diretório corrente**

### **mkdir – (make directory) Criando diretórios.**

### **rm – (remove) Removendo arquivos e/ou diretórios.**

\$rm -r nomedodiretorio Remove diretórios não vazios

\$rm -f Remove os arquivos sem perguntar.

### **touch – Criando arquivos vazios**

### **cp – (copy) Copiando arquivos**

nome-original nome-novo

### **mv – (move) Movendo ou Renomeando arquivos. Copia um arquivo com um novo nome e deleta o antigo.**

nome-original nome-novo

### **echo – comando que permite mostrar mensagens na tela.**

### **cat – comando que permite listar o conteúdo de um arquivo.**

O carácter > (**maior**) ou >>(**maior maior**) redireciona (>) ou anexa (>>) a saída de um comando para um arquivo texto.

### **Exemplo**

echo abacaxi

echo abacaxi > TESTE

cat TESTE

**O que apareceu na tela?** \_\_\_\_\_

echo abacate > TESTE  
cat TESTE  
O que apareceu na tela? \_\_\_\_\_

echo abacaxi >> TESTE  
E agora, o que apareceu na tela? \_\_\_\_\_

## Formas de navegar dentro dos diretórios:

**Caminho absoluto:** Todo caminho absoluto inicia no diretório raiz (/), por exemplo: /home/aluno, a partir do qual, todos caminhos absolutos derivam, formando uma árvore de diretórios. A vantagem dos caminhos absolutos é poder identificar arquivos.

**Caminho relativo:** Todo caminho relativo não contém uma 'barra' no início. A referência é geralmente o diretório atual (pwd) onde o shell se encontra .

Em caminhos relativos é possível utilizar outros atalhos como:

- `~ til`, que referenciam o diretório pessoal. Por exemplo, `ls ~/Downloads` lista a pasta Downloads da pasta pessoal do usuário atual.
- `.` ponto, que significa o diretório atual. Por exemplo, `./run-app`, executa o arquivo `run-app` que está localizado no diretório atual.
- `..` dois-pontos, que significa o diretório pai. Por exemplo, `cd ..`, muda para o diretório pai.

A vantagem dos caminhos relativos é poder executar comandos mais curtos. Por exemplo, é mais fácil executar o aplicativo `run-app` a partir do diretório atual com o comando `./run-app` do que executá-lo com o comando absoluto `/usr/local/bin/run-app`.

### Exemplos

`/home/bohlke/programas/testes/arquivol.blah`

é um caminho absoluto para o arquivo. Pode-se alcançar o `arquivol.blah` independentemente do diretório atual.

`programas/teste/arquivol.blah`

é um caminho relativo para o mesmo arquivo. Só se pode alcançá-lo se estivermos previamente em `/home/bohlke`

### Utilize:

- **man** *comando* e descubra o que o comando faz.
- **clear** para limpar a tela
- **history** para mostrar todos os comandos digitados anteriormente.
- A tecla **TAB** para **autocompletar** o nome de arquivos/diretórios.
- A tecla “**seta para cima**” e “**seta para baixo**” para verificar os últimos comandos digitados.