

Computação Aplicada a Meteorologia

Curso de Linux

O termo Linux é usado em vários contextos com significados diferentes. A rigor, Linux é um kernel. No entanto, em alguns contextos, Linux significa sistema operacional (não qualquer sistema operacional, mas um que use o kernel Linux).

Sistema Operacional: é um software que serve de interface entre o computador e o usuário, gerenciando recursos (como memória, processamento etc.).

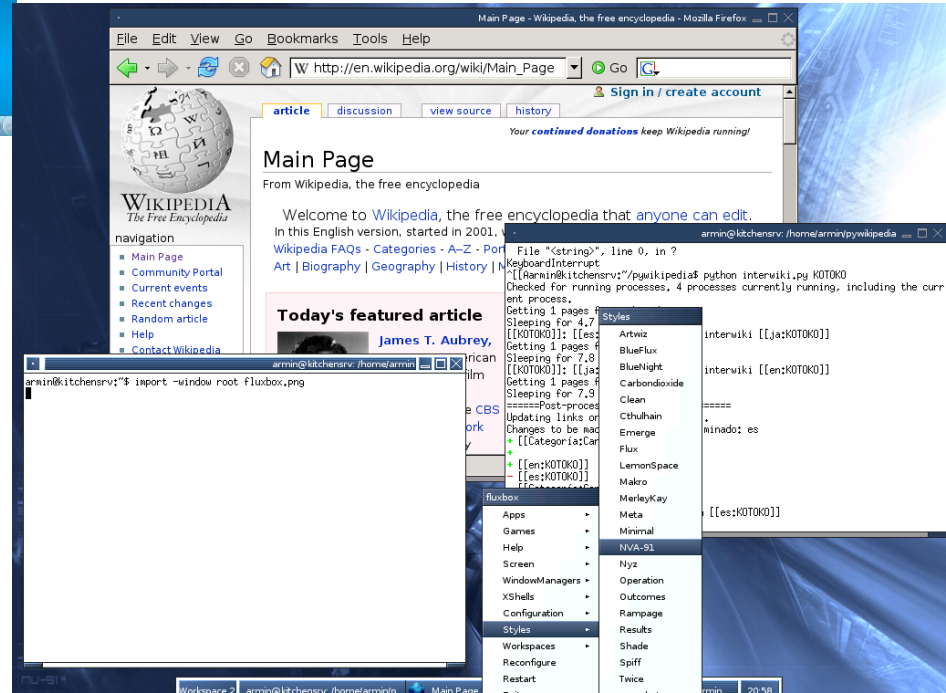
Kernel: é o núcleo ou cerne do sistema operacional (é a parte deste que fica mais próxima" do hardware).

Computação Aplicada a Meteorologia

Distribuições



Ambiente Grafico



Computação Aplicada a Meteorologia

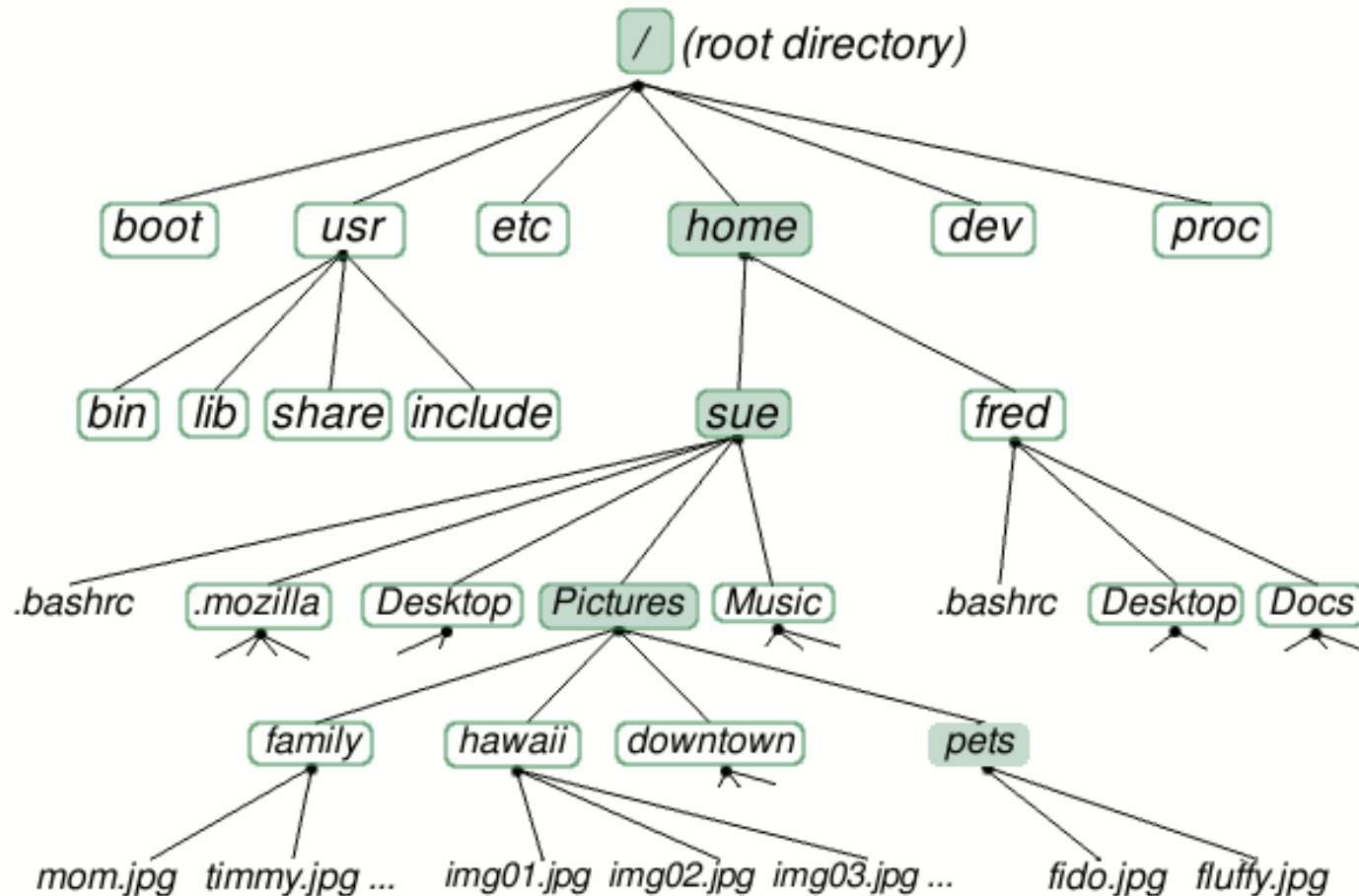
Aplicativos

Descrição	Programas usados no Windows	Programas usados no GNU/Linux
Pacote Office	Microsoft Office	KOffice, OpenOffice
Processador de Texto	Microsoft Word	KWrite, OpenOffice Writer
Planilhas	Excel	KSpread, OpenOffice Calc
Apresentações	PowerPoint	KPresenter, OpenOffice Impress
E-mail	Outlook	Evolution
Gravação de mídia	Nero	Brasero, K3b
IDEs LaTeX	TeXnicCenter, WinEdit	Kile, Texmaker
Compactadores de arquivos	Winrar, Winzip	ark, bzip2, tar
Leitor de PDF	Adobe Reader	Adobe Reader, Evince, Kpdf
Modelagem 3D	3D Studio MAX, Blender, Maya	Blender, K-3D, Maya
Players de vídeo	Windows Media Player	Kaffeine, MPlayer Totem, VLC
Players de música	iTunes, Winamp, Windows Media Player	Amarok, Audacious, RhythmBox
Edição de vídeos	Windows Movie Maker	Cinelerra, Kino
Clientes P2P BitTorrent	μ Torrent, Azureus	Azureus, KTorrent, Transmission
Mensageiros instantâneos	MSN	aMSN, Kopete, Pidgin
Browser	Firefox, Google Chrome, Microsoft Internet Explorer, Opera	Firefox, Galeon, Google Chrome, Konqueror, Opera

Computação Aplicada a Meteorologia

Diretórios e arquivos

Visão geral da organização dos arquivos no Linux



No topo da hierarquia de arquivos fica o chamado diretório raiz (ou, mais apropriadamente, diretório root), pois a estrutura de diretórios é chamada também de "Árvore de Diretórios".

Computação Aplicada a Meteorologia

Diretórios e arquivos

Visão geral da organização dos arquivos no Linux

Diretorio root /

Este é o diretório principal do sistema. Dentro dele estão todos os diretórios do sistema. O diretório root é representado por uma barra (/).

Diretório /bin

Contem comandos e programas essenciais para todos os usuários (alguns desses comandos serão tratados no próximo capítulo).

Diretório /boot

Contem arquivos necessários para a inicialização do sistema.

Diretório /dev

Dispositivos: o /dev contem referências para todos os dispositivos, os quais são representados como arquivos com propriedades especiais.

Computação Aplicada a Meteorologia

Diretórios e arquivos

Diretorio /etc

Contem arquivos de configuração.

Diretorio /home

Contem os diretórios dos usuários.

Diretorio /lib

Contem bibliotecas (que são subprogramas ou códigos auxiliares utilizados por programas) essenciais para o funcionamento do Linux, e também os módulos do kernel.

Diretorio /media

Este diretório contem subdiretórios que são usados como pontos de montagem para medias removíveis, como disquetes, cdroms, pen drives etc.

Computação Aplicada a Meteorologia

Diretórios e arquivos

Diretorio /root

Diretório "home" do super. usuário (usuário root). Não confundir com o diretório root, o /. O diretório **/root** contém os arquivos do usuário root. O diretório / é o topo da hierarquia de arquivos.

Usuário root: É o administrador do sistema, possui acesso a todos os comandos e arquivos.

Diretorio /tmp

Para arquivos temporários.

Diretorio /usr

Contém programas, bibliotecas etc.

Computação Aplicada a Meteorologia

Diretórios e arquivos

Diretorio /usr/bin

E onde ficam os binários (executáveis) de programas não-essenciais (os essenciais ficam no /bin).

Diretorio /usr/src

Código-fonte.

Diretorio /var

Contem arquivos “variáveis”, como logs, base de dados.

Diretorio /var/log

Como o próprio nome diz, possui arquivos de log.

Arquivo de log: E um arquivo que armazena registros de eventos relevantes de um programa ou do sistema.

Computação Aplicada a Meteorologia

Diretórios e arquivos

Diretorio /var/run

Contem informação sobre a execução do sistema desde a sua ultima inicialização.

Existem outros diretórios previstos no padrão, mas, por enquanto, estes já são suficientes.

Computação Aplicada a Meteorologia

Modo texto ou (Modo de linha de comando)

Como dito anteriormente, não é apenas pelo modo gráfico que o usuário consegue interagir com o sistema.

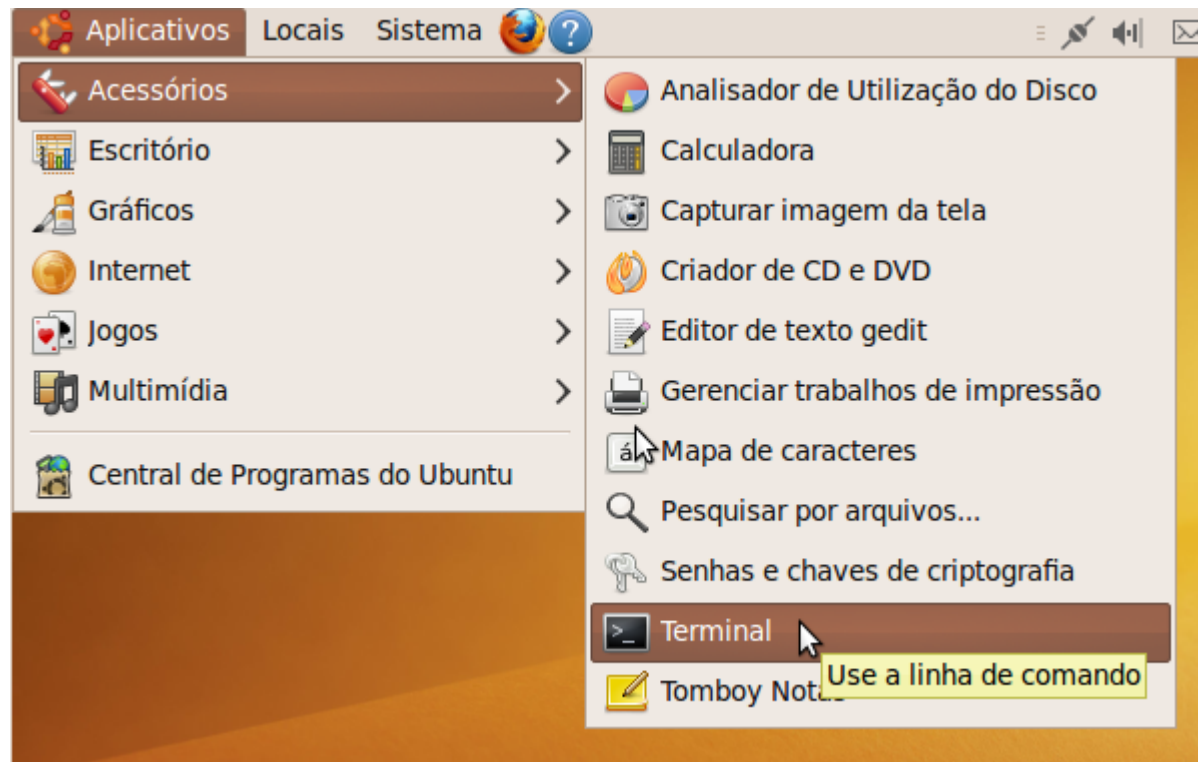
É possível fazer isso pelo modo texto, digitando comandos e nomes de programas para conseguir uma “resposta” do sistema.

Por isso, o modo texto é também chamado de linha de comando.

Computação Aplicada a Meteorologia

Modo texto ou (Modo de linha de comando)

Uma forma de usar o modo texto ou linha de comando e usar um “emulador de terminal”, isto é, dentro do modo gráfico, abre-se um programa que funciona como linha de comando. Para fazer isso no ambiente GNOME, vá em Aplicativos) Acessórios) Terminal.



Computação Aplicada a Meteorologia

Modo texto ou (Modo de linha de comando)

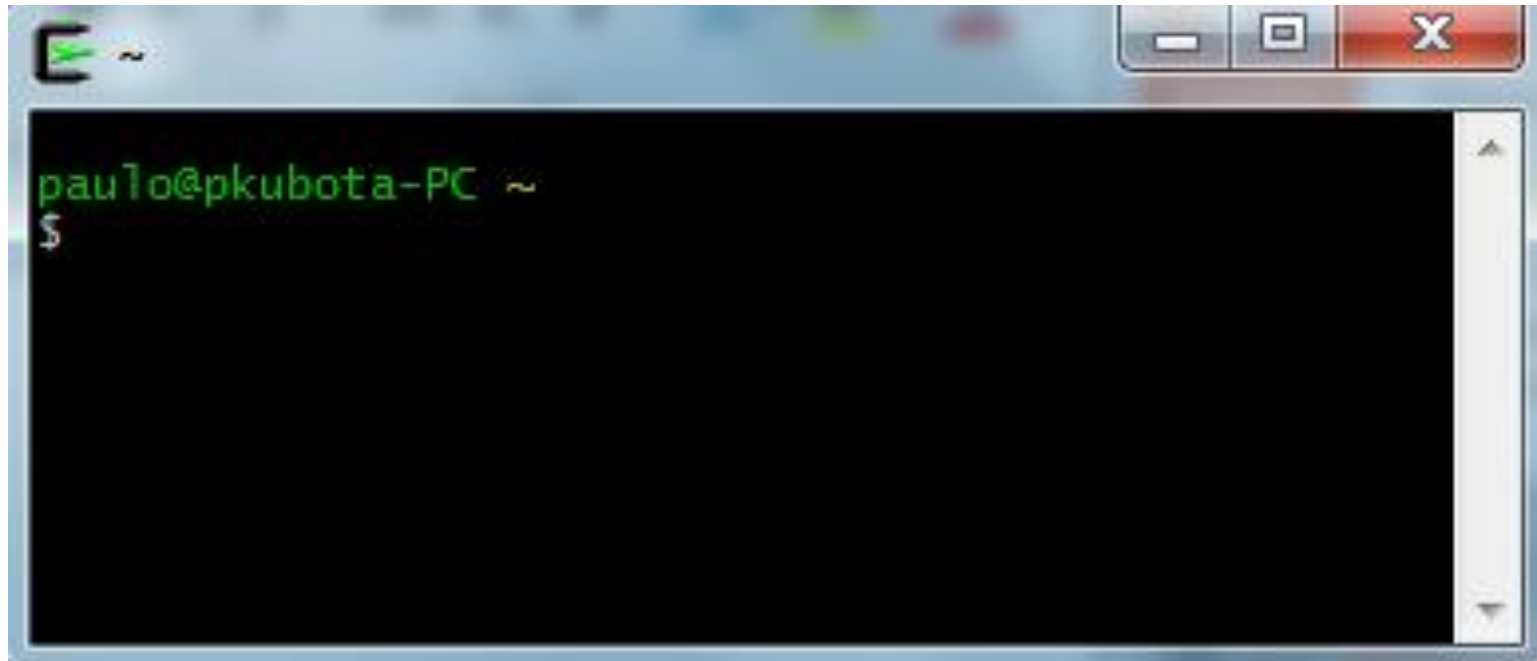
Prompt

O prompt do BASH tem a seguinte aparência:

username@nomedamaquina:diretorio\$

No caso de

curso@curso-desktop:~\$



Computação Aplicada a Meteorologia

Modo texto ou (Modo de linha de comando)

Sintaxe dos comandos

É importante lembrar que a linha de comando é case sensitiva, isto é, diferencia letras maiúsculas de minúsculas. Portanto, “echo” é diferente de “Echo”, que são diferentes de “ECHO”. Isso também vale para nomes de diretórios e arquivos.

comando -opção1 -opção2 --opção3 argumento

Outro ponto importante é que você pode digitar os comandos e nomes de arquivos ou diretórios pela metade e depois pressionar a tecla “Tab”. Ele irá completar o nome para você.

Computação Aplicada a Meteorologia

- Comandos para navegação no sistema de arquivos

Cmd	Descrição	Argumentos Principais	
<code>pwd</code>	Mostra o caminho do dir. atual		
<code>ls</code>	Lista arquivos e pastas do diretório atual	-a ou --all -l ou --list -h	Lista arquivos ocultos Exibe detalhes Human readable
<code>tree</code>	Lista uma árvore de arquivos e pastas cuja raiz é a diretório atual	-d -L level	Lista apenas os diretórios omitindo Lista no máximo level níveis da árv.
<code>cd</code>	Muda de diretório		

Atalhos especiais:

~ : Diretório home do usuário

. : Diretório atual

.. : Diretório pai do diretório atual

Computação Aplicada a Meteorologia

Comandos para gerenciamento de arquivos e pastas

Cmd	Descrição	Exemplos	
mkdir	Cria um diretório	\$ mkdir dir1	Cria dir1
cp	Copia arquivo ou pasta	\$ cp ~/arq1 /bin	Copia arq1 para /bin
mv	Move arquivo ou pasta	\$ mv ~/arq1 /bin	Move arq1 para /bin
rm	Remove arquivo ou pasta	\$ rm ~/arq1	Remove arq1
touch	Modifica o time stamp de um arquivo, criando-o por padrão caso não exista	\$ touch a	Cria o arquivo a, ou modifica a timestamp do arquivo a para a data e hora atuais
ln	Cria um link para um arquivo	\$ ln -s /home/user/arquivo.txt link_name	Cria o link link_name no diretório atual que aponta para /home/usr/arquivo.txt

Computação Aplicada a Meteorologia

Como visualizar um arquivo pelo Shell ?

Cmd	Descrição
<code>cat</code>	Imprime o arquivo na saída padrão (stdout)
<code>less</code>	Pagina o arquivo na saída e exibe-o na saída padrão, oferecendo a possibilidade de navegar pelo arquivo.
<code>more</code>	Equivalente ao <code>less</code> , com menos recursos.
<code>wc</code>	Conta o número de linhas, caracteres e bytes de um arquivo

Entrada e Saída de Programas Linux

- Stdout ou Saída Padrão Ex: Tela, Impressora, Arquivo
- Stdin ou Entrada Padrão Ex: Teclado, Arquivo, Pipe
- Stderr ou Saída de Erro Padrão Ex: Tela, Arquivo

Computação Aplicada a Meteorologia

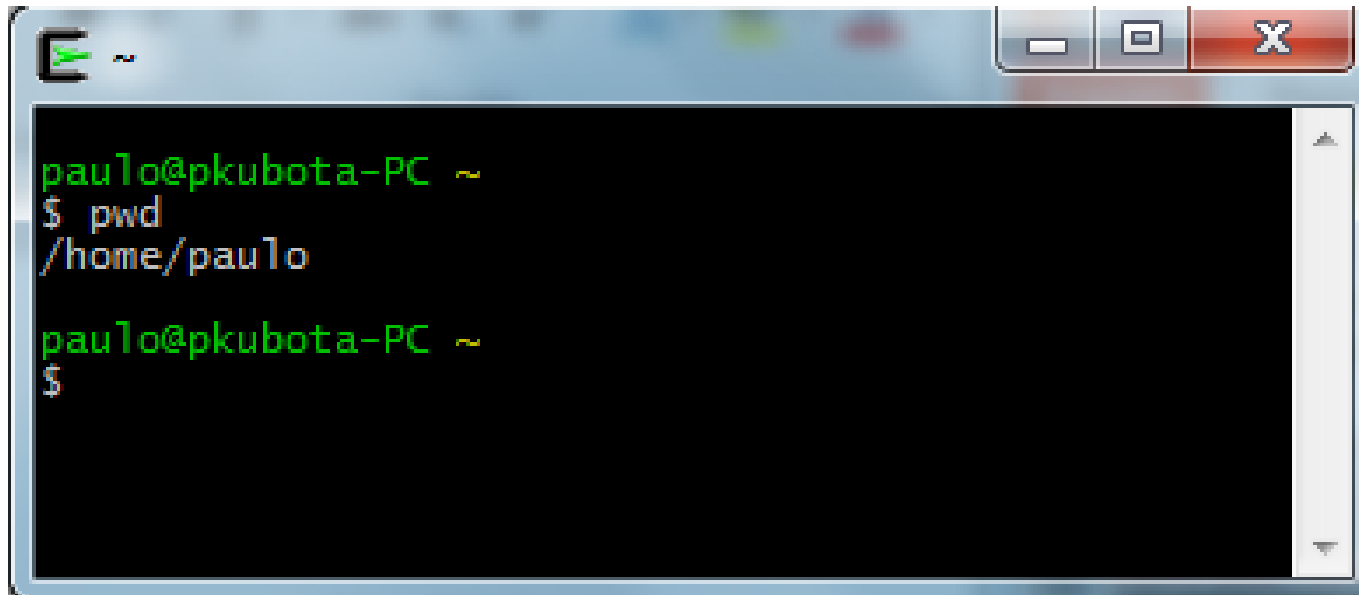
pwd (print working directory)

Mostra o nome e o **caminho** do diretório atual (**diretório em que o usuário esta**).

```
$ pwd
```

```
/home/curso
```

```
$
```



```
paulo@pkubota-PC ~  
$ pwd  
/home/paulo  
paulo@pkubota-PC ~  
$
```

Computação Aplicada a Meteorologia

ls (**list**)

Sintaxe básica:

ls [opções] [diretorio1] [diretorio2] ...

Exemplos

1. O comando abaixo lista os diretórios e arquivos do /

\$ ls /

2. O comando abaixo lista os diretórios e arquivos do **/etc**

\$ ls /etc

3. Para listar o conteúdo do / e do **/etc**, de uma só vez, use:

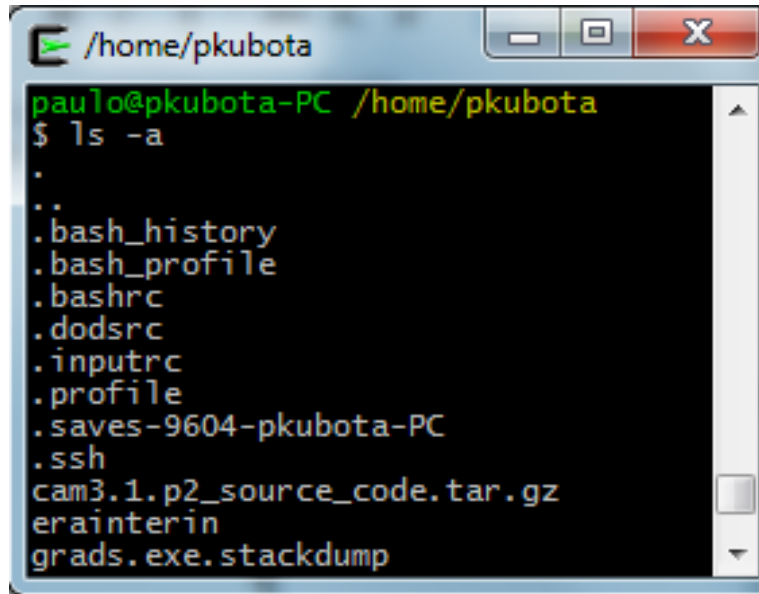
\$ ls / /etc

Computação Aplicada a Meteorologia

ls (**list**)

Sintaxe básica:

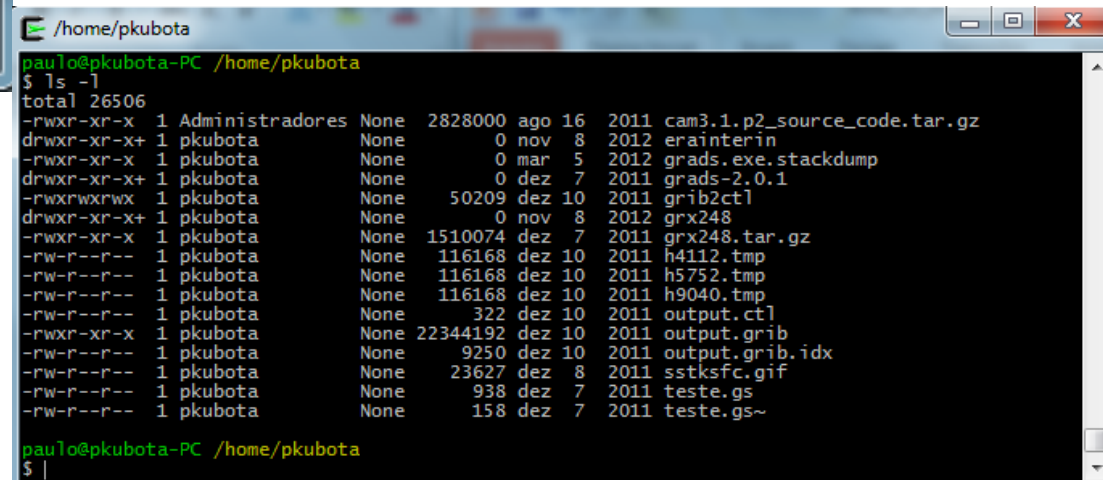
ls [opções] [diretorio1] [diretorio2] ...



```
paulo@pkubota-PC /home/pkubota
$ ls -a
.
..
.bash_history
.bash_profile
.bashrc
.dodsrc
.inputrc
.profile
.save-9604-pkubota-PC
.ssh
cam3.1.p2_source_code.tar.gz
erainterin
grads.exe.stackdump
```

* -a = mostra arquivos ocultos

* -l = mostra bytes, permissões, diretório, etc



```
paulo@pkubota-PC /home/pkubota
$ ls -l
total 26506
-rwxr-xr-x 1 Administradores None 2828000 ago 16 2011 cam3.1.p2_source_code.tar.gz
drwxr-xr-x+ 1 pkubota None 0 nov 8 2012 erainterin
-rwxr-xr-x 1 pkubota None 0 mar 5 2012 grads.exe.stackdump
drwxr-xr-x+ 1 pkubota None 0 dez 7 2011 grads-2.0.1
-rwxrwxrwx 1 pkubota None 50209 dez 10 2011 grib2ctl
drwxr-xr-x+ 1 pkubota None 0 nov 8 2012 grx248
-rwxr-xr-x 1 pkubota None 1510074 dez 7 2011 grx248.tar.gz
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h4112.tmp
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h5752.tmp
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h9040.tmp
-rw-r--r-- 1 pkubota None 322 dez 10 2011 output.ct1
-rwxr-xr-x 1 pkubota None 22344192 dez 10 2011 output.grib
-rw-r--r-- 1 pkubota None 9250 dez 10 2011 output.grib.idx
-rw-r--r-- 1 pkubota None 23627 dez 8 2011 sstksfc.gif
-rw-r--r-- 1 pkubota None 938 dez 7 2011 teste.gs
-rw-r--r-- 1 pkubota None 158 dez 7 2011 teste.gs~

paulo@pkubota-PC /home/pkubota
$ |
```

Computação Aplicada a Meteorologia

cd (**change directory**)

Entra em um diretório.

Sintaxe básica:

cd [**diretório**]

1. Para entrar no diretório **root**, use

\$ cd /

2. Para entrar no diretório **/tmp**, basta usar o seguinte comando

\$ cd **/tmp**

3. Para subir um diretório acima, use:

\$ cd **..**

4. Para voltar ao diretório imediatamente anteriormente acessado, basta usar:

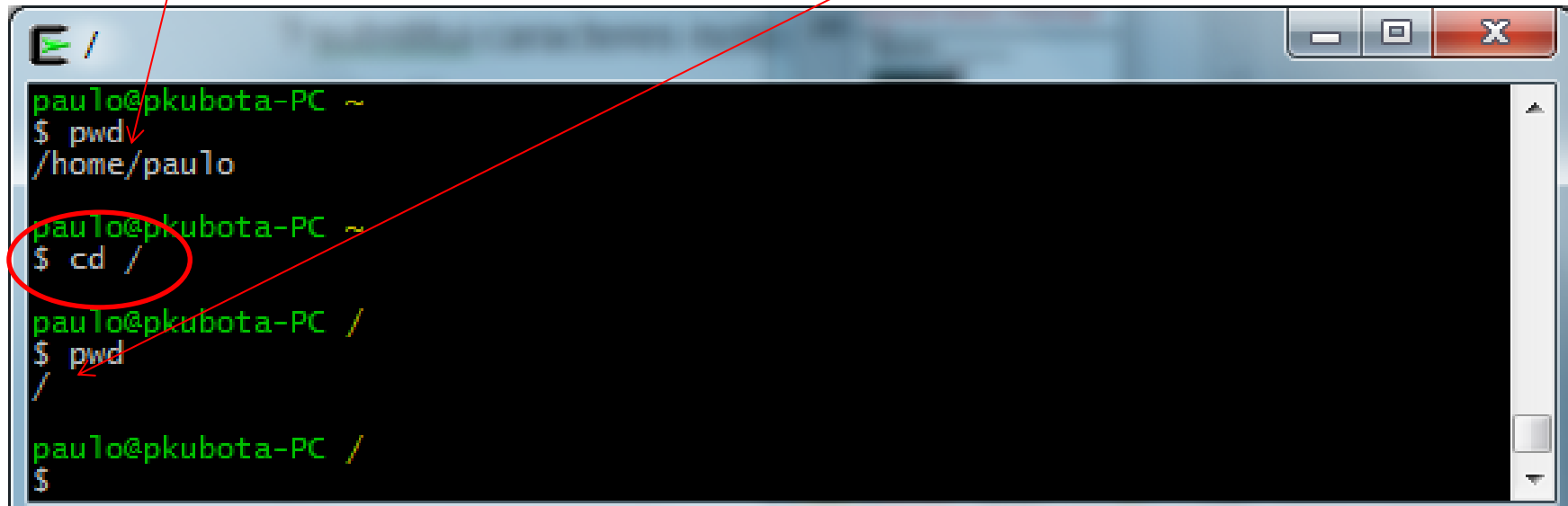
\$ cd **-**

Computação Aplicada a Meteorologia

cd (change directory)

Diretório home

Diretório root



```
paulo@pkubota-PC ~  
$ pwd  
/home/paulo  
paulo@pkubota-PC ~  
$ cd /  
paulo@pkubota-PC /  
$ pwd  
/  
paulo@pkubota-PC /  
$
```

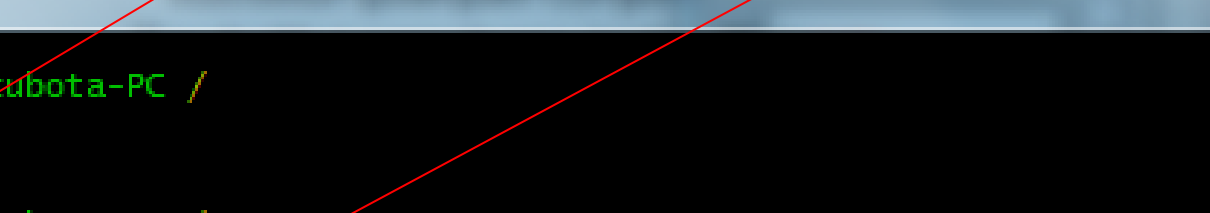
The image shows a terminal window with a black background and green text. The window title bar includes a green icon and the text 'E /'. The terminal output shows the user 'paulo' at host 'pkubota-PC' in the home directory '~'. The first command is 'pwd', which outputs '/home/paulo'. The second command is 'cd /', which is circled in red. The third command is 'pwd', which outputs '/'. A red arrow points from the 'Diretório root' label to the '/' output. The window has standard Linux window controls (minimize, maximize, close) in the top right corner.

Computação Aplicada a Meteorologia

cd (change directory)

Diretório root

Diretório home



```
paulo@pkubota-PC /
$ pwd
/
paulo@pkubota-PC /
$ cd -
/home/paulo
paulo@pkubota-PC ~
$ pwd
/home/paulo
paulo@pkubota-PC ~
$ |
```

Computação Aplicada a Meteorologia

mkdir (**make directory**)

Cria novos diretórios (vazios).

Sintaxe básica:

\$ mkdir [caminho1/diretorio1] [caminho2/diretorio2] ...

Exemplos

1. Para criar os diretórios “**Pasta1**” e “**Pasta2**” dentro do diretório **/tmp**, fazemos:

```
$ mkdir /tmp/Pasta1 /tmp/Pasta2
```

2. Naturalmente, se estivéssemos dentro do diretório **/tmp**, não seria necessário usar o caminho absoluto:

```
$ pwd
```

```
/tmp
```

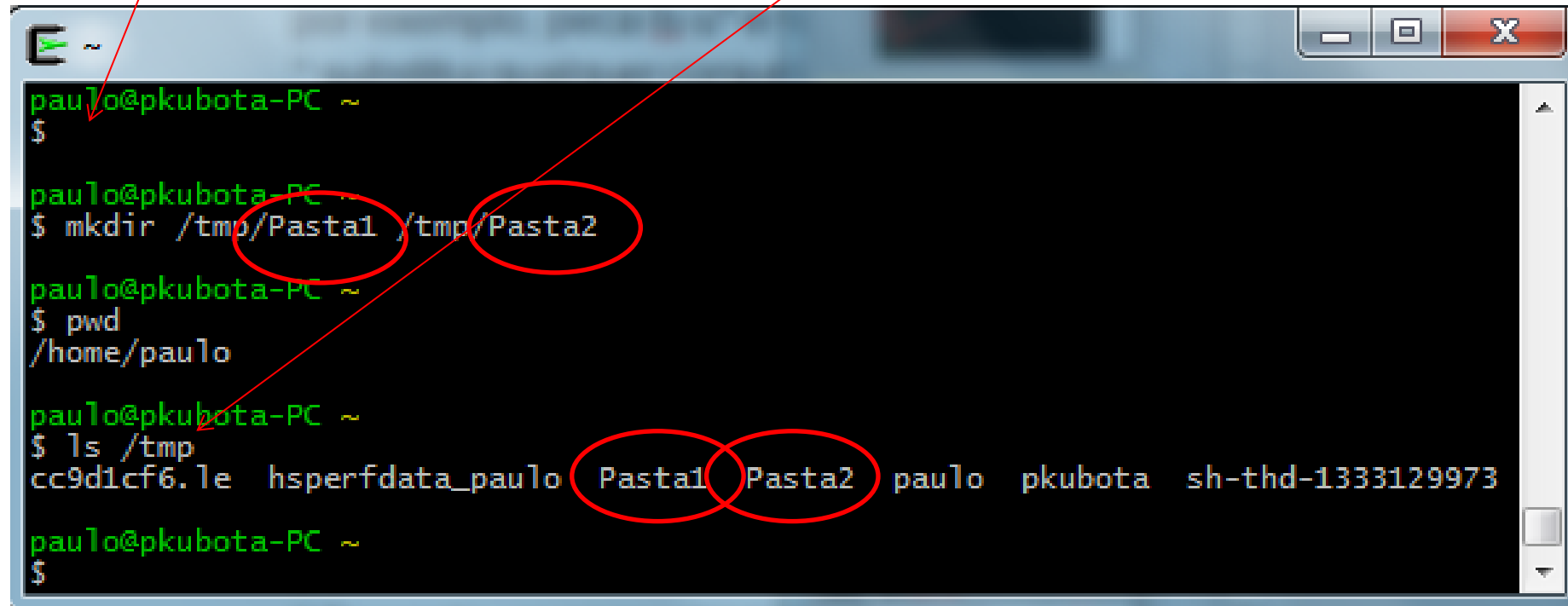
```
$ mkdir Pasta1 Pasta2
```


Computação Aplicada a Meteorologia

mkdir (**make directory**)

Diretório home

Diretório tmp



```
paulo@pkubota-PC ~  
$  
paulo@pkubota-PC ~  
$ mkdir /tmp/Pasta1 /tmp/Pasta2  
paulo@pkubota-PC ~  
$ pwd  
/home/paulo  
paulo@pkubota-PC ~  
$ ls /tmp  
cc9d1cf6.1e  hspcrfdata_paulo  Pasta1  Pasta2  paulo  pkubota  sh-thd-1333129973  
paulo@pkubota-PC ~  
$
```

The terminal window shows the execution of the `mkdir` command to create two directories, `Pasta1` and `Pasta2`, in the `/tmp` directory. The `pwd` command confirms the current directory is `/home/paulo`. The `ls /tmp` command lists the contents of the `/tmp` directory, showing the newly created directories among other files. Red circles highlight the paths `/tmp/Pasta1` and `/tmp/Pasta2` in the `mkdir` command, and the directory names `Pasta1` and `Pasta2` in the `ls` output. Red arrows point from the labels 'Diretório home' and 'Diretório tmp' to the tilde (~) and /tmp paths respectively.

Computação Aplicada a Meteorologia

rmdir (**remove directory**)

Remove um ou mais diretórios vazios.

Sintaxe básica:

```
$ rmdir [caminho1/diretorio1] [caminho2/diretorio2] ...
```

Exemplos

1. Para remover os diretórios “**Pasta1**” e “**Pasta2**” criados como nos exemplos do comando mkdir, poderíamos usar:

```
$ rmdir /tmp/Pasta1 /tmp/Pasta2
```

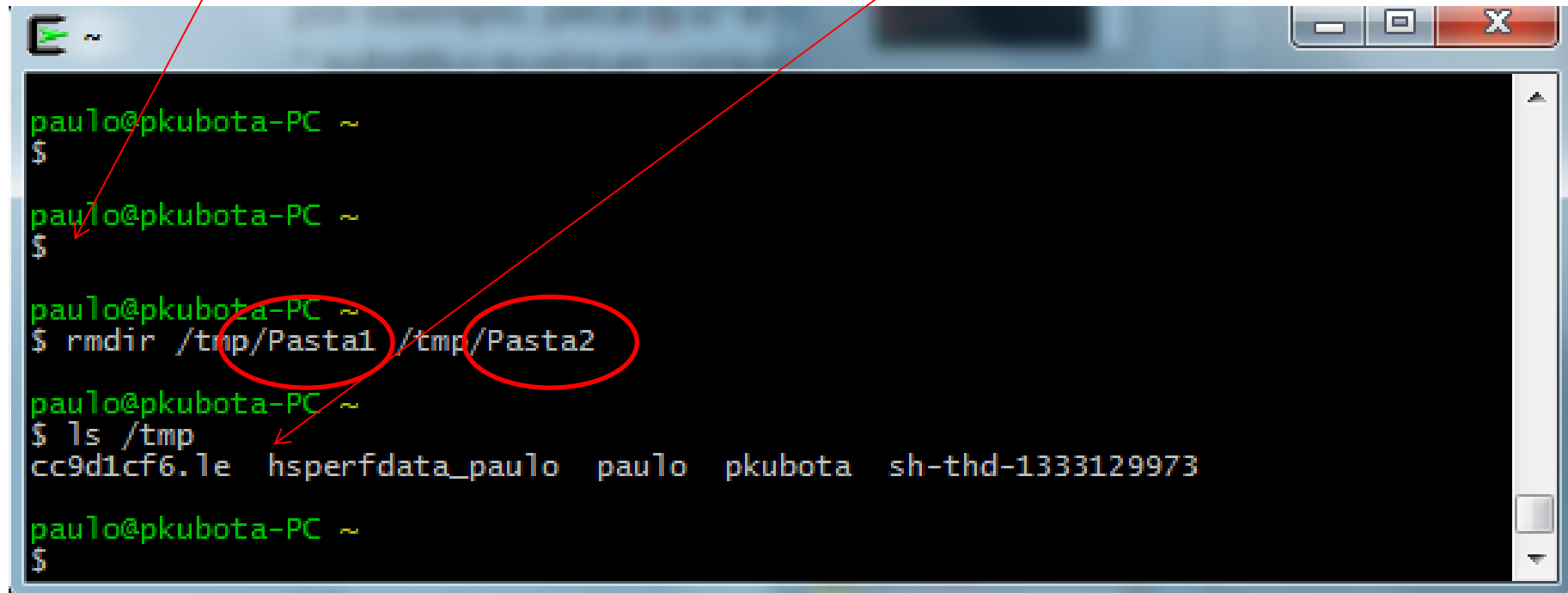
Computação Aplicada a Meteorologia

rmmdir (**remove directory**)

Remove um ou mais diretórios vazios.

Diretório home

Diretório tmp



A terminal window titled "E ~" with standard window controls (minimize, maximize, close) in the top right corner. The terminal shows a series of commands and their outputs. The prompt is "paulo@pkubota-PC ~". The first two prompts are followed by a dollar sign "\$" without any output. The third prompt is followed by the command `$ rmmdir /tmp/Pasta1 /tmp/Pasta2`, where the paths `/tmp/Pasta1` and `/tmp/Pasta2` are circled in red. The fourth prompt is followed by the command `$ ls /tmp`, and the output is `cc9d1cf6.1e hspcrfdata_paulo paulo pkubota sh-thd-1333129973`. The final prompt is followed by a dollar sign "\$" without any output. Two red arrows point from text boxes above to the terminal: one from "Diretório home" to the tilde "~" in the prompt, and another from "Diretório tmp" to the `/tmp` part of the `rmmdir` command.

```
paulo@pkubota-PC ~
$
paulo@pkubota-PC ~
$
paulo@pkubota-PC ~
$ rmmdir /tmp/Pasta1 /tmp/Pasta2
paulo@pkubota-PC ~
$ ls /tmp
cc9d1cf6.1e  hspcrfdata_paulo  paulo  pkubota  sh-thd-1333129973
paulo@pkubota-PC ~
$
```

Computação Aplicada a Meteorologia

touch (**Touch time**)

Pode ser usado para **criar** novos arquivos vazios e também para **mudar** a data e a hora de criação de arquivos existentes.

Sintaxe básica:

touch [opções] [arquivo1] [arquivo2] ...

Opções

. -t [[YY]YY]MMDDhhmm[.ss] - Altera a data e hora do arquivo para o ano YYYY (nesse caso, pode-se usar os quatro dígitos ou apenas dois), para o mês MM, para o dia DD, para a hora hh, para o minuto mm e para o segundo ss. Lembrando que as opções de ano e segundo são opcionais (por isso foram colocadas entre colchetes).

Exemplos

1. Para criar um arquivo vazio chamado "**arquivonovo**" no diretório atual, poderíamos usar:

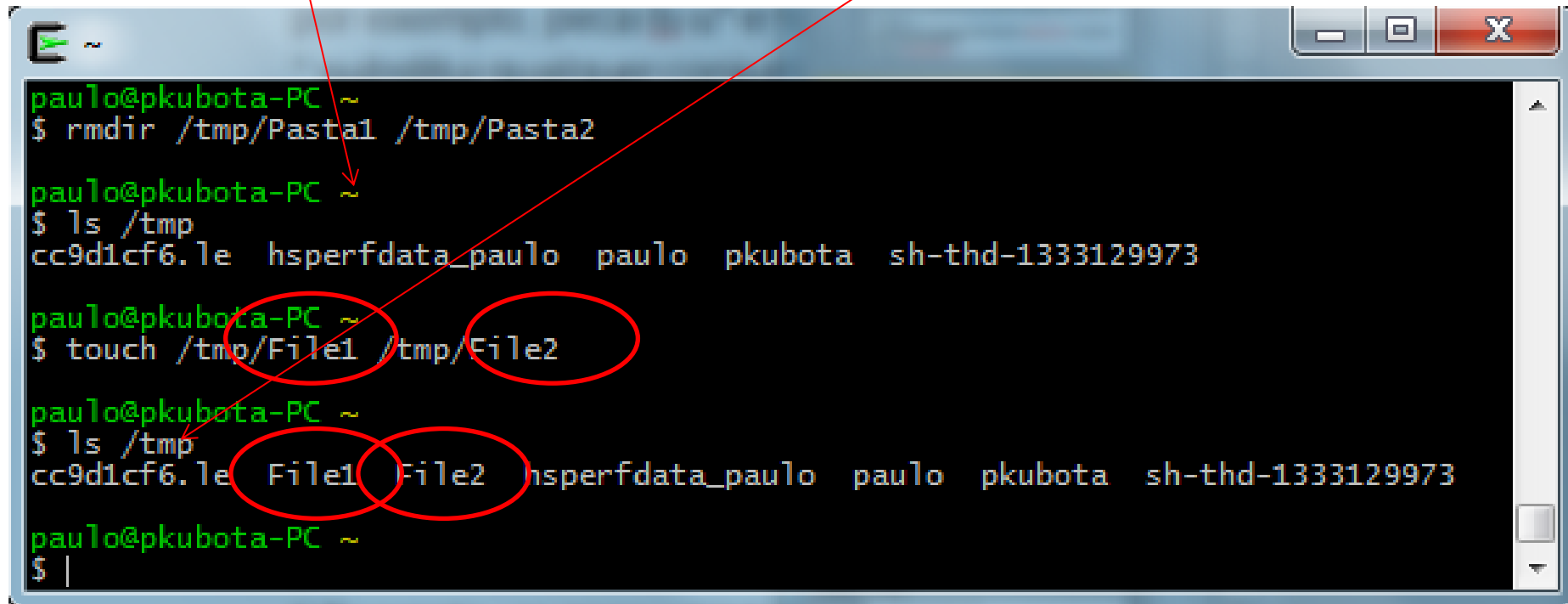
\$ touch **arquivonovo**

Computação Aplicada a Meteorologia

touch (Touch time)

Diretório home

Diretório tmp



A terminal window titled "paulo@pkubota-PC" showing a series of commands and their outputs. The window has a standard Linux terminal interface with a title bar and window controls. The commands and outputs are as follows:

```
paulo@pkubota-PC ~  
$ rmdir /tmp/Pasta1 /tmp/Pasta2  
  
paulo@pkubota-PC ~  
$ ls /tmp  
cc9d1cf6.1e  hspcrfdata_paulo  paulo  pkubota  sh-thd-1333129973  
  
paulo@pkubota-PC ~  
$ touch /tmp/File1 /tmp/File2  
  
paulo@pkubota-PC ~  
$ ls /tmp  
cc9d1cf6.1e  File1  File2  hspcrfdata_paulo  paulo  pkubota  sh-thd-1333129973  
  
paulo@pkubota-PC ~  
$ |
```

Annotations in the image include:

- A red arrow pointing from the "Diretório home" label to the tilde (~) in the prompt "paulo@pkubota-PC ~".
- A red arrow pointing from the "Diretório tmp" label to the "/tmp" in the command "ls /tmp".
- Red circles highlighting the paths "/tmp/File1" and "/tmp/File2" in the "touch" command.
- Red circles highlighting the files "File1" and "File2" in the output of the "ls /tmp" command.

Computação Aplicada a Meteorologia

touch (Touch time)

Data atual

```
paulo@pkubota-PC ~  
$ ls -l /tmp/File*  
-rw-r--r-- 1 paulo None 0 set 19 12:07 /tmp/File1  
-rw-r--r-- 1 paulo None 0 set 19 12:07 /tmp/File2  
  
paulo@pkubota-PC ~  
$ |
```

touch -t 201301010000 /tmp/File1 /tmp/File2

Data modificada

```
paulo@pkubota-PC ~  
$ touch -t 201301010000 /tmp/File1 /tmp/File2  
  
paulo@pkubota-PC ~  
$ ls -l /tmp/File*  
-rw-r--r-- 1 paulo None 0 jan  1  2013 /tmp/File1  
-rw-r--r-- 1 paulo None 0 jan  1  2013 /tmp/File2  
  
paulo@pkubota-PC ~  
$
```

Computação Aplicada a Meteorologia

rm (**remove**)

Remove arquivos e diretórios.

Sintaxe básica:

\$ rm [opções] [arquivo1] [arquivo2] ...

Exemplos:

1. Criamos um arquivo chamado “**teste**” no diretório **/tmp**:

\$ touch **/tmp/teste**

Agora vamos removê-lo:

\$ rm /tmp/teste

Opções

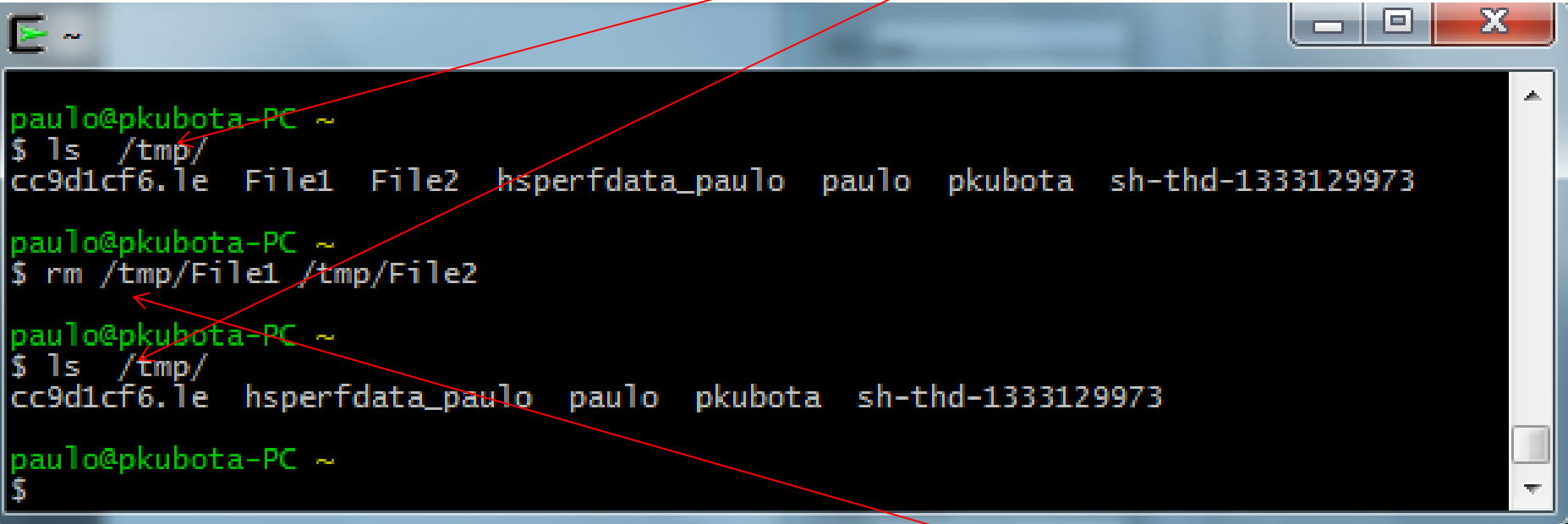
. **-r**: Opção usada para remover recursivamente diretórios e seu conteúdo. Pode ser usada também para remover diretórios vazios.

Computação Aplicada a Meteorologia

rm (**remove**)

Remove arquivos e diretórios.

Lista os arquivos no diretório /tmp

A terminal window titled "E ~" with standard window controls (minimize, maximize, close) in the top right corner. The terminal shows a series of commands and their outputs. The first command is "ls /tmp/" which lists several files including "cc9d1cf6.1e", "File1", "File2", "hsperfdata_paulo", "paulo", "pkubota", and "sh-thd-1333129973". The second command is "rm /tmp/File1 /tmp/File2". The third command is another "ls /tmp/" which now only shows "cc9d1cf6.1e", "hsperfdata_paulo", "paulo", "pkubota", and "sh-thd-1333129973". The prompt "\$" is shown at the end of the last line. Two red arrows originate from the text box above: one points to the "File1" and "File2" in the first "ls" output, and the other points to the "rm" command.

```
paulo@pkubota-PC ~  
$ ls /tmp/  
cc9d1cf6.1e  File1  File2  hsperfdata_paulo  paulo  pkubota  sh-thd-1333129973  
  
paulo@pkubota-PC ~  
$ rm /tmp/File1 /tmp/File2  
  
paulo@pkubota-PC ~  
$ ls /tmp/  
cc9d1cf6.1e  hsperfdata_paulo  paulo  pkubota  sh-thd-1333129973  
  
paulo@pkubota-PC ~  
$
```

Remove os arquivos no diretório /tmp

Computação Aplicada a Meteorologia

cp (**copy**)

Este comando serve para copiar arquivos.

Sintaxe básica:

\$ cp [opções] [origem] [destino]

Exemplos

1. Para copiar o arquivo “**teste**” do **/tmp** para o diretório home do usuário:

\$ cp **/tmp/teste** ~

Opções

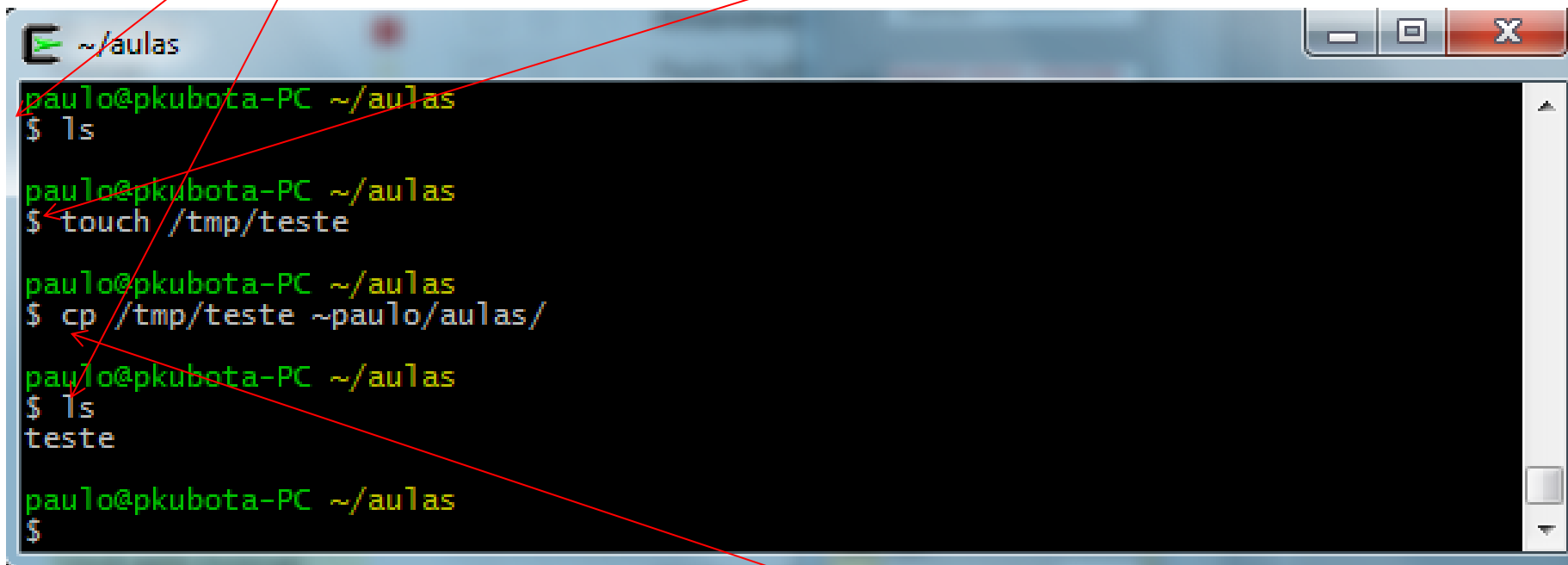
. **-R**: Copia recursivamente os subdiretórios e seu conteúdo.

Computação Aplicada a Meteorologia

cp (**copy**)

Lista os arquivos no diretório **/home/paulo/aulas**

Cria o arquivo **teste** no diretório **/tmp**



```
~ /aulas
paulo@pkubota-PC ~/aulas
$ ls
paulo@pkubota-PC ~/aulas
$ touch /tmp/teste
paulo@pkubota-PC ~/aulas
$ cp /tmp/teste ~/paulo/aulas/
paulo@pkubota-PC ~/aulas
$ ls
teste
paulo@pkubota-PC ~/aulas
$
```

The image shows a terminal window titled '~ /aulas' with standard Linux window controls (minimize, maximize, close) in the top right. The terminal text is as follows: paulo@pkubota-PC ~/aulas, \$ ls, paulo@pkubota-PC ~/aulas, \$ touch /tmp/teste, paulo@pkubota-PC ~/aulas, \$ cp /tmp/teste ~/paulo/aulas/, paulo@pkubota-PC ~/aulas, \$ ls, teste, paulo@pkubota-PC ~/aulas, \$. Three red arrows originate from the text boxes above: one points to the 'ls' command, another points to the 'touch' command, and a third points to the 'cp' command.

copiam arquivo **teste** do diretório **/tmp** para o diretório **/home/paulo/aulas/**

Computação Aplicada a Meteorologia

mv (**move**)

Move e renomeia arquivos e diretórios.

Sintaxe básica

```
$ mv [opções] [origem] [destino]
```

1. Suponha que um usuário possui um arquivo em sua home chamado **arquivo1**. Para renomear este arquivo para **arquivonovo**, supondo que o usuário está em sua home, bastaria usar:

```
$ mv arquivo1 arquivonovo
```

Opções

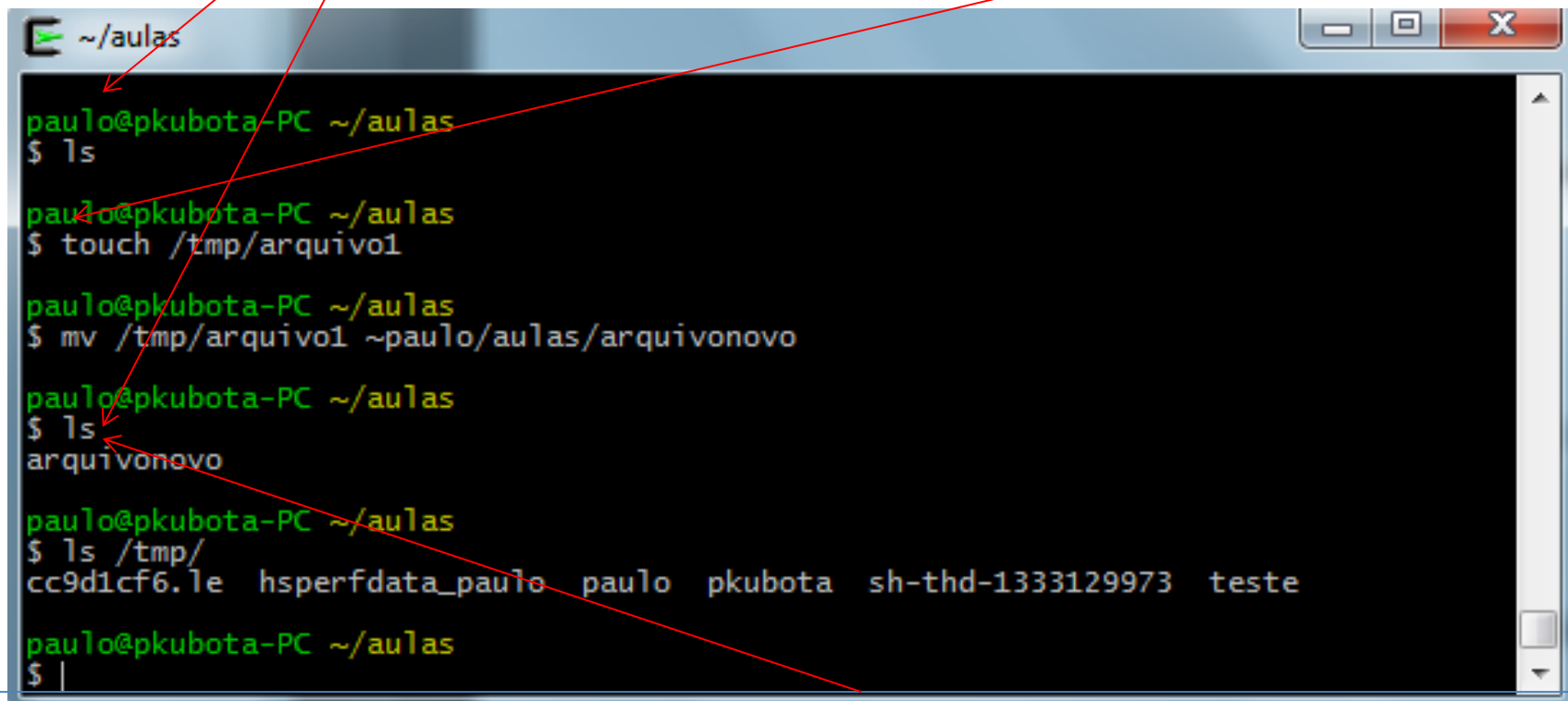
. **-r**: Como outros comandos, essa opção move diretórios e seu conteúdo recursivamente.

Computação Aplicada a Meteorologia

mv (**move**)

Lista os arquivos no diretório **/home/paulo/aulas**

Cria o arquivo **arquivo1** no diretório **/tmp**



```
~/aulas
paulo@pkubota-PC ~/aulas
$ ls
paulo@pkubota-PC ~/aulas
$ touch /tmp/arquivo1
paulo@pkubota-PC ~/aulas
$ mv /tmp/arquivo1 ~/paulo/aulas/arquivonovo
paulo@pkubota-PC ~/aulas
$ ls
arquivonovo
paulo@pkubota-PC ~/aulas
$ ls /tmp/
cc9d1cf6.1e  hsperfddata_paulo  paulo  pkubota  sh-thd-1333129973  teste
paulo@pkubota-PC ~/aulas
$
```

Move o arquivo **arquivo1** do diretório **/tmp** para o diretório **/home/paulo/aulas/**

Computação Aplicada a Meteorologia

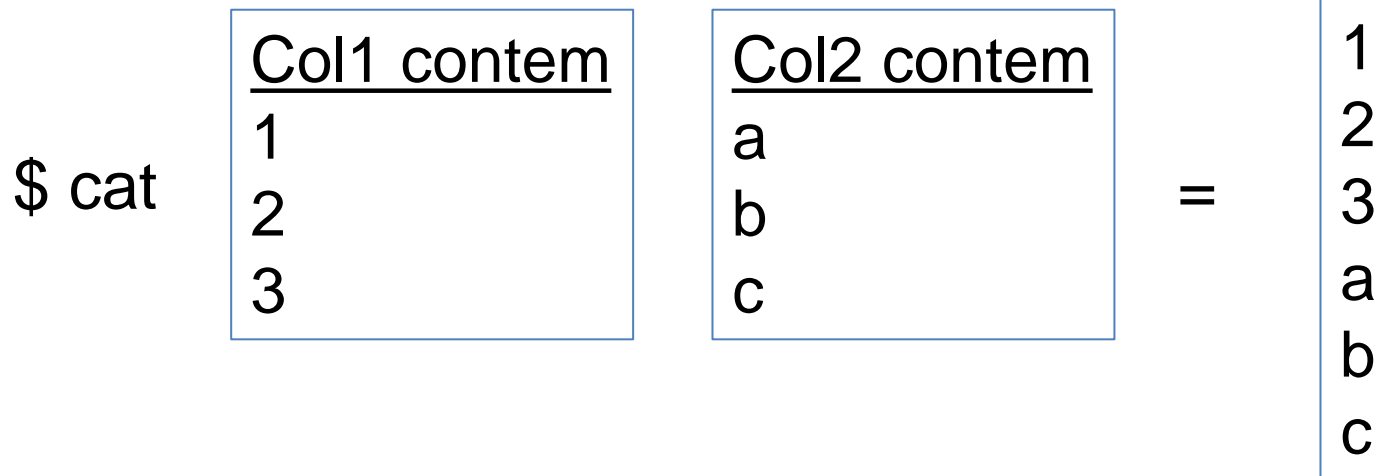
cat (concatenate)

Concatena arquivos e imprime o resultado no terminal.

Sintaxe básica

\$ cat [arquivo1] [arquivo2] ...

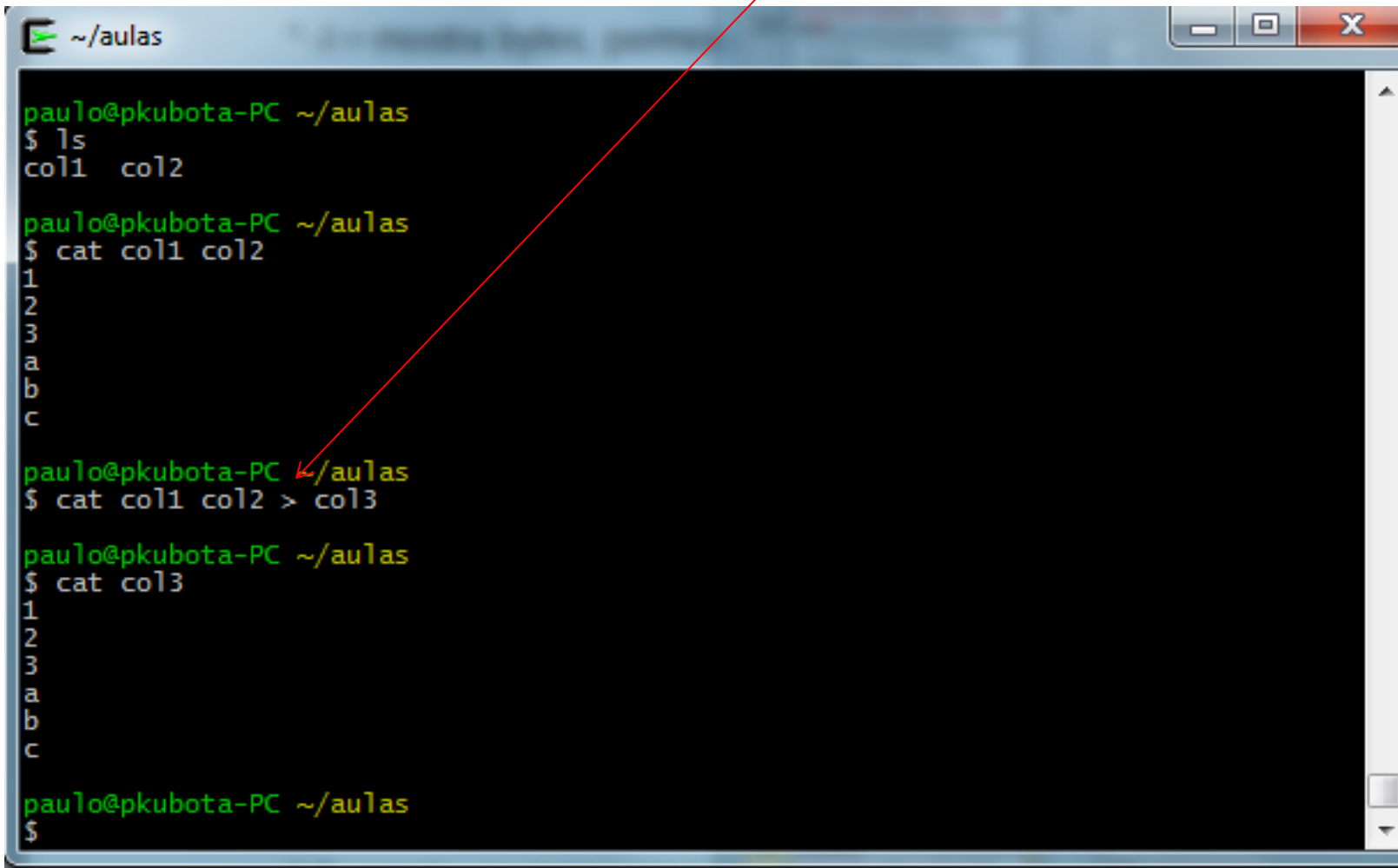
\$cat col1 col2



Computação Aplicada a Meteorologia

cat (concatenate)

Direciona (>) o resultado do comando **cat** para arquivo **col3**



```
~/aulas  
paulo@pkubota-PC ~/aulas  
$ ls  
col1  col2  
paulo@pkubota-PC ~/aulas  
$ cat col1 col2  
1  
2  
3  
a  
b  
c  
paulo@pkubota-PC ~/aulas  
$ cat col1 col2 > col3  
paulo@pkubota-PC ~/aulas  
$ cat col3  
1  
2  
3  
a  
b  
c  
paulo@pkubota-PC ~/aulas  
$
```

A terminal window titled '~/aulas' showing a series of commands and their outputs. The user 'paulo' is at the 'pkubota-PC' machine. The first command is 'ls', which lists 'col1' and 'col2'. The second command is 'cat col1 col2', which outputs the contents of both files: '1', '2', '3', 'a', 'b', and 'c' on separate lines. The third command is 'cat col1 col2 > col3', which concatenates the contents of 'col1' and 'col2' into a new file 'col3'. A red arrow points from the text 'Direciona (>) o resultado do comando cat para arquivo col3' to the '>' symbol in this command. The fourth command is 'cat col3', which outputs the contents of 'col3', showing the same sequence of lines as before. The prompt '\$' is shown at the end of the last line.

Computação Aplicada a Meteorologia

find (find)

O comando **find** é usado para procurar por diretórios e arquivos no disco. Possui varias opções, mas mostraremos apenas alguns exemplos simples.

Sintaxe básica

```
$ find [diretório origem] -name [arquivo]
```

Exemplos

1. Este exemplo procura por um **arquivo** ou **diretório** com o nome "**Documents**" a partir do (/) (**diretório root**):

```
$ find / -name Documents
```

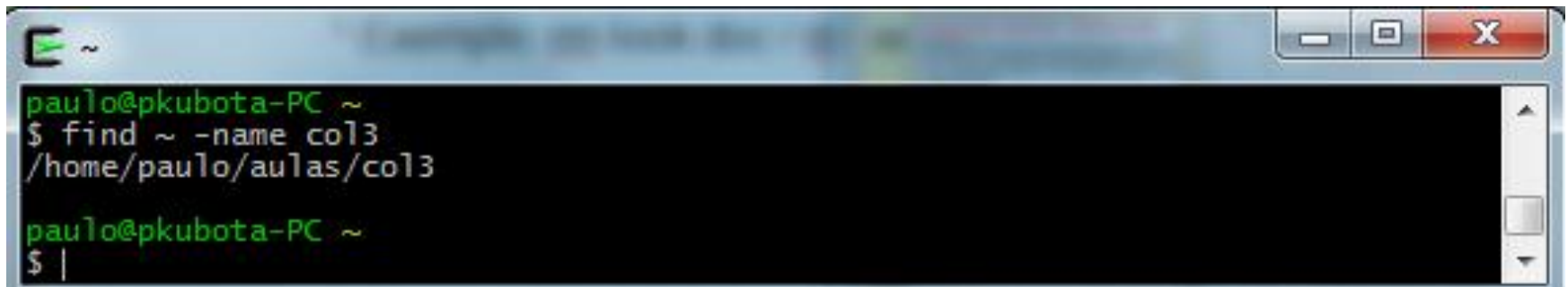
2. Este outro procura por um **arquivo** ou **diretório** com o nome "**Music**" a partir do diretório **home** (~) do usuário:

```
$ find ~ -name Music
```

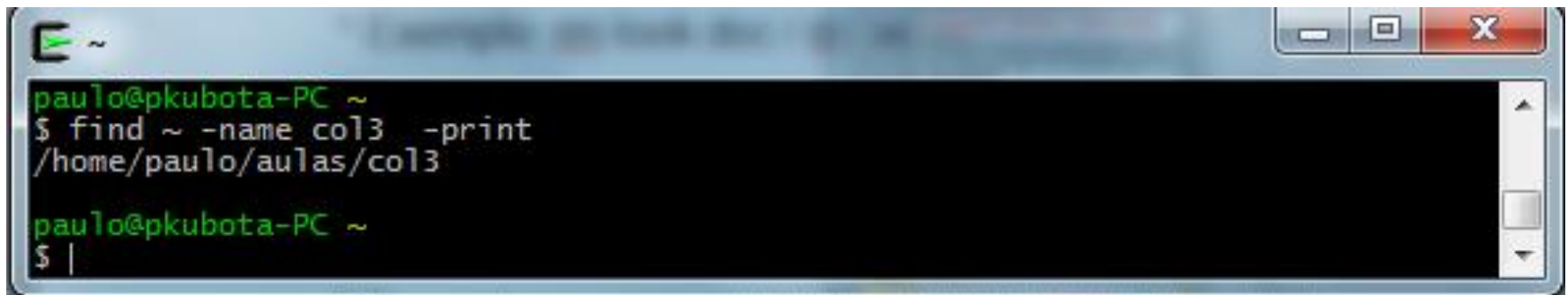
Computação Aplicada a Meteorologia

find (find)

É importante salientar que “a partir do diretório x” significa que o comando procurará dentre tudo o que estiver contido no tal **diretório (x)**, incluindo os **arquivos** e os **subdiretórios**, bem como seu **conteúdo** e assim por diante.

A terminal window with a title bar containing a green icon and a tilde (~). The prompt is 'paulo@pkubota-PC ~'. The command '\$ find ~ -name col3' is entered, and the output is '/home/paulo/aulas/col3'. The prompt returns to '\$ |' on a new line.

```
paulo@pkubota-PC ~  
$ find ~ -name col3  
/home/paulo/aulas/col3  
  
paulo@pkubota-PC ~  
$ |
```

A terminal window with a title bar containing a green icon and a tilde (~). The prompt is 'paulo@pkubota-PC ~'. The command '\$ find ~ -name col3 -print' is entered, and the output is '/home/paulo/aulas/col3'. The prompt returns to '\$ |' on a new line.

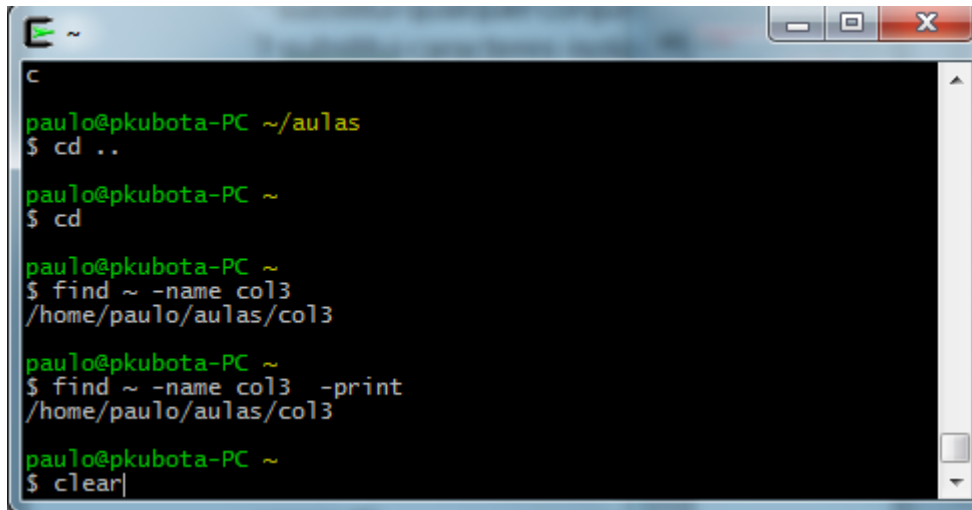
```
paulo@pkubota-PC ~  
$ find ~ -name col3 -print  
/home/paulo/aulas/col3  
  
paulo@pkubota-PC ~  
$ |
```


Computação Aplicada a Meteorologia

clear (clear)

Use o comando clear e descubra o que ele faz:

\$ clear



```
paulo@pkubota-PC ~/aulas
$ cd ..

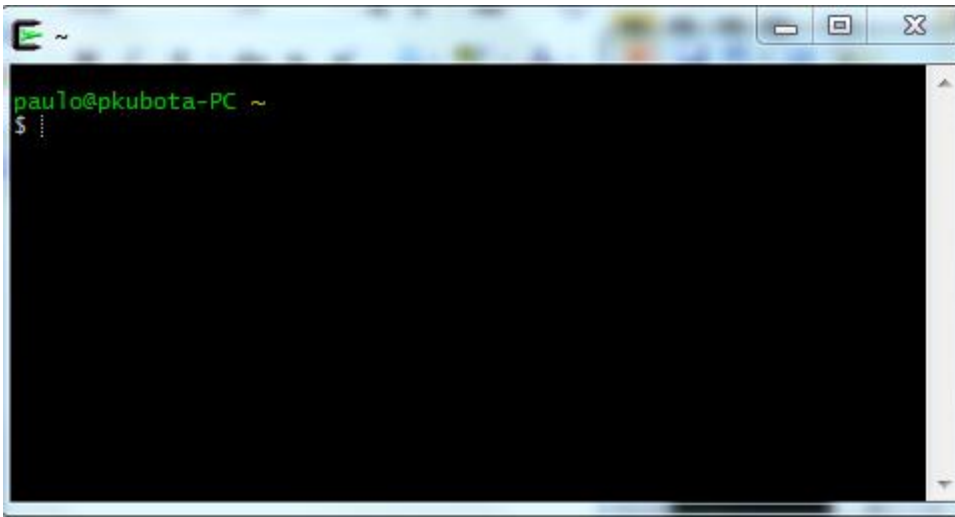
paulo@pkubota-PC ~
$ cd

paulo@pkubota-PC ~
$ find ~ -name col3
/home/paulo/aulas/col3

paulo@pkubota-PC ~
$ find ~ -name col3 -print
/home/paulo/aulas/col3

paulo@pkubota-PC ~
$ clear
```

\$ clear



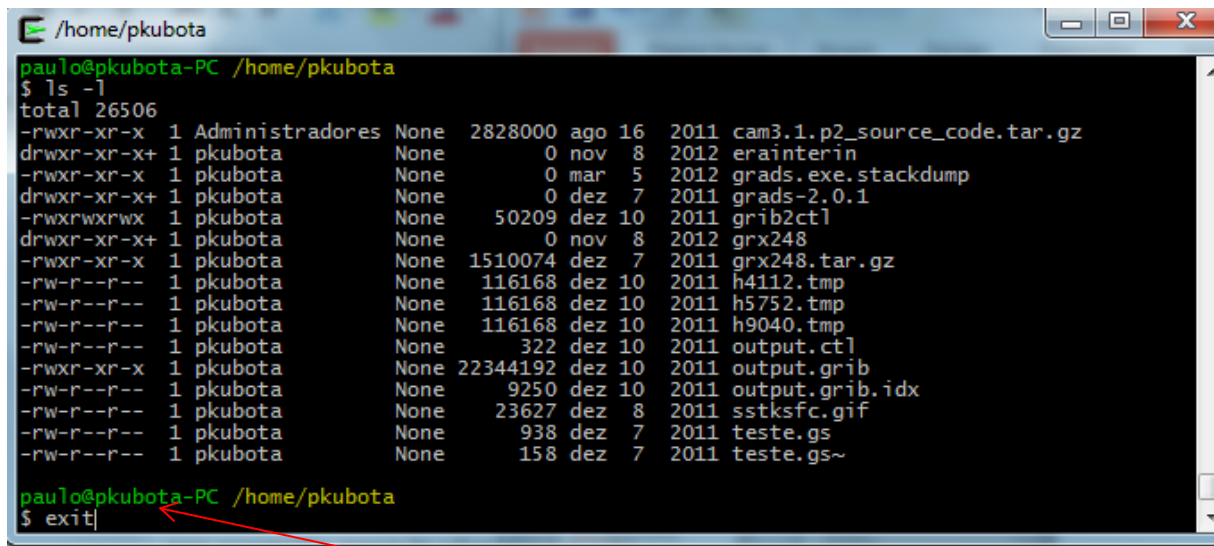
```
paulo@pkubota-PC ~
$
```

**Limpa as
informações o
terminal**

Computação Aplicada a Meteorologia

exit (exit)

Este comando serve para sair do shell (interpretador) e para efetuar o log out do usuário no terminal.

A screenshot of a terminal window titled '/home/pkubota'. The prompt is 'paulo@pkubota-PC /home/pkubota'. The user has entered '\$ ls -l', and the terminal displays a long list of files with their permissions, owner, size, and date. At the bottom, the user has entered '\$ exit'. A red arrow points from the text 'Fecha o terminal' in the block below to the 'exit' command.

```
paulo@pkubota-PC /home/pkubota
$ ls -l
total 26506
-rwxr-xr-x 1 Administradores None 2828000 ago 16 2011 cam3.1.p2_source_code.tar.gz
drwxr-xr-x+ 1 pkubota None 0 nov 8 2012 erainterin
-rwxr-xr-x 1 pkubota None 0 mar 5 2012 grads.exe.stackdump
drwxr-xr-x+ 1 pkubota None 0 dez 7 2011 grads-2.0.1
-rwxrwxrwx 1 pkubota None 50209 dez 7 2011 grib2ctl
drwxr-xr-x+ 1 pkubota None 0 nov 8 2012 grx248
-rwxr-xr-x 1 pkubota None 1510074 dez 7 2011 grx248.tar.gz
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h4112.tmp
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h5752.tmp
-rw-r--r-- 1 pkubota None 116168 dez 10 2011 h9040.tmp
-rw-r--r-- 1 pkubota None 322 dez 10 2011 output.ct1
-rwxr-xr-x 1 pkubota None 22344192 dez 10 2011 output.grib
-rw-r--r-- 1 pkubota None 9250 dez 10 2011 output.grib.idx
-rw-r--r-- 1 pkubota None 23627 dez 8 2011 sstksfc.gif
-rw-r--r-- 1 pkubota None 938 dez 7 2011 teste.gs
-rw-r--r-- 1 pkubota None 158 dez 7 2011 teste.gs~

paulo@pkubota-PC /home/pkubota
$ exit
```

Fecha o terminal

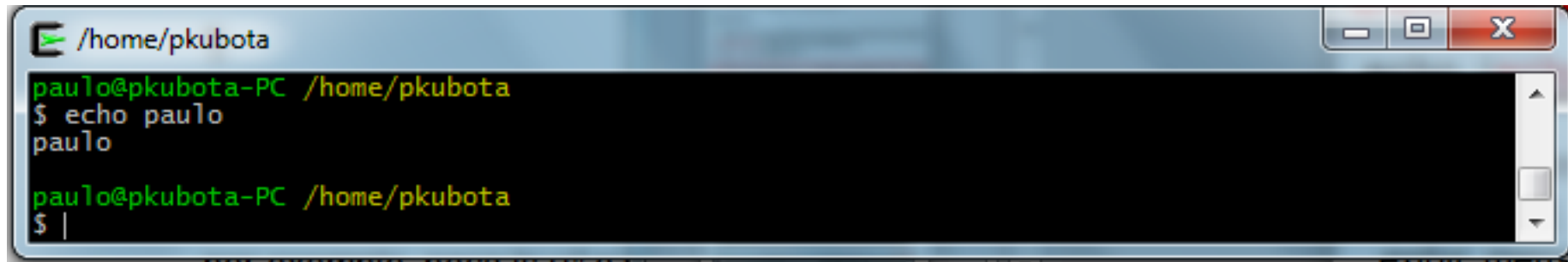
Computação Aplicada a Meteorologia

echo (echo)

Mostra um texto. Por agora, pode parecer um comando pouco útil, mas é bastante usado sobretudo em scripts para exibir mensagens ao usuário.

Sintaxe básica

\$ echo mensagem

A screenshot of a Linux terminal window. The title bar shows the path "/home/pkubota" and standard window controls. The terminal content shows a user named "paulo" at a machine named "pkubota-PC" in the directory "/home/pkubota". The user enters the command "\$ echo paulo", and the terminal outputs "paulo". The prompt "\$" is shown again on the next line, followed by a vertical bar "|" indicating the cursor is ready for input.

```
/home/pkubota
paulo@pkubota-PC /home/pkubota
$ echo paulo
paulo
paulo@pkubota-PC /home/pkubota
$ |
```

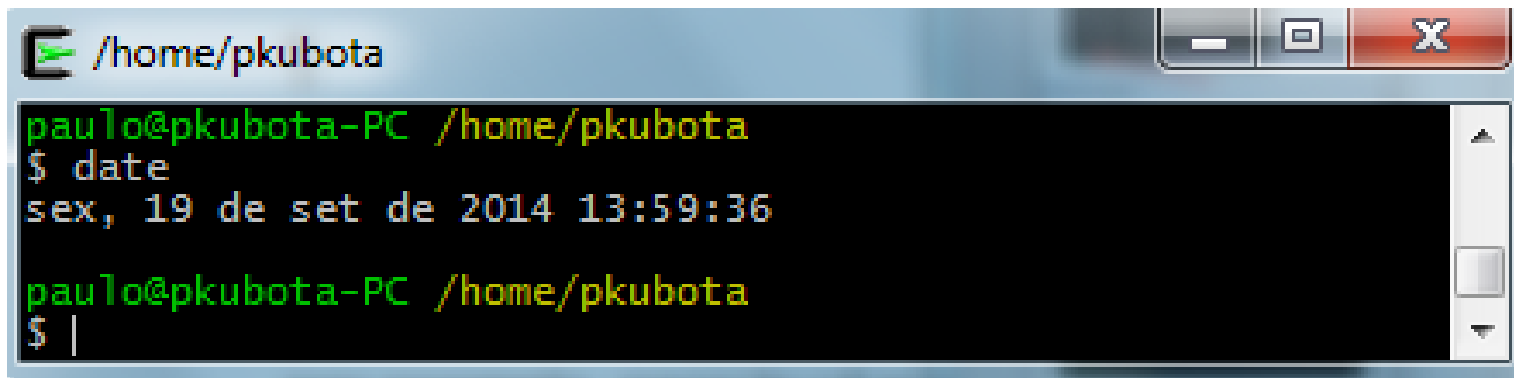
Computação Aplicada a Meteorologia

date (**date**)

O comando date **imprime** ou **modifica** a **data** e o **horário** do sistema. É importante salientar que somente o usuário root e usuários privilegiados podem rodar este comando.

Sintaxe básica:

\$ date [data]

A screenshot of a Linux terminal window. The title bar shows the file manager icon and the path /home/pkubota. The terminal content shows a user named paulo at a machine named pkubota-PC, in the directory /home/pkubota. The user enters the command '\$ date', and the output is 'sex, 19 de set de 2014 13:59:36'. The prompt returns to '\$ |' for the next command.

```
/home/pkubota
paulo@pkubota-PC /home/pkubota
$ date
sex, 19 de set de 2014 13:59:36
paulo@pkubota-PC /home/pkubota
$ |
```

Computação Aplicada a Meteorologia

date (**date**)

Exemplos

1. Para visualizar a data e a hora do sistema:

```
$ date
```

```
Mon Mar 8 14:45:21 BRT 2010
```

2. Para **alterar a data e a hora do sistema**, basta usar o comando da seguinte maneira:

```
$ date MMDDhhmm[[YYyy][.ss]]
```

Onde **MM** é o mês, **DD** é o dia, **hh** é a hora, **mm** são os minutos. Opcionalmente, podem ser usados o **ano** (com 2 ou 4 dígitos) e os segundos (**ss**). Para alterar a data do sistema para o dia 1 de fevereiro e o horário para 14:30, poderíamos fazer:

```
$ date 02011430
```

Computação Aplicada a Meteorologia

chmod (**change mode**)

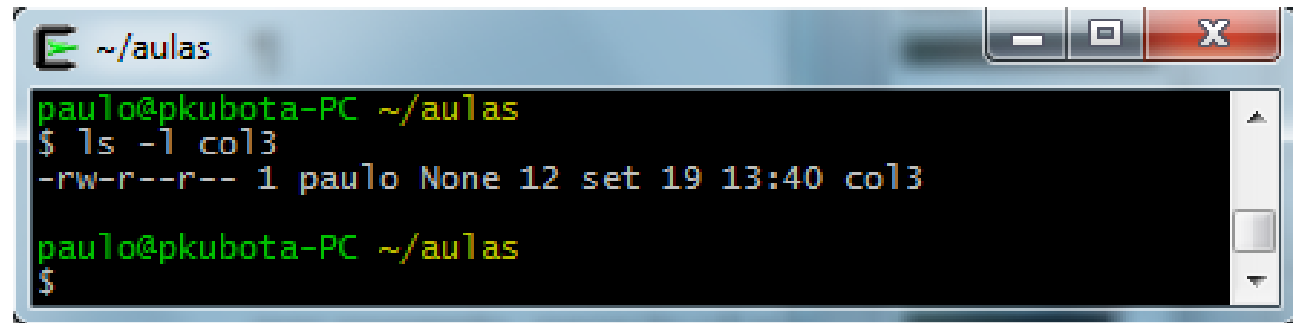
Este comando é usado para **mudar** permissões de arquivos ou **diretórios**.

Sintaxe básica:

\$ chmod [permissões] [diretório/arquivo]

Ha duas formas de usar o comando.

- 1) Você precisa saber que “**u**” representa o dono (“**user**”), “**g**”, o **grupo**, “**o**”, os **demaís usuários** e “**a**”, por sua vez, representa **todos** (“**all**”).



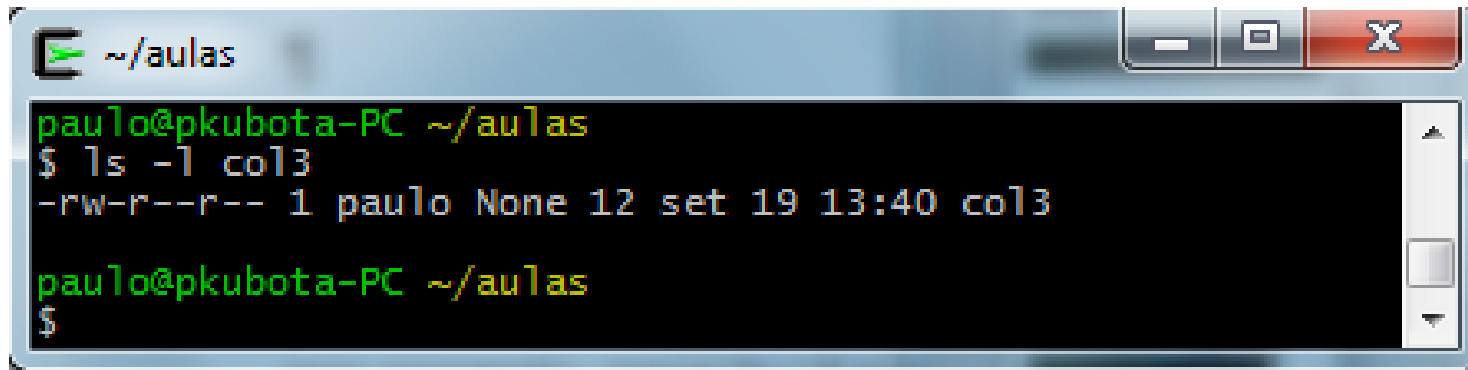
```
~/aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 12 set 19 13:40 col3
paulo@pkubota-PC ~/aulas
$
```

-rw-r--r-- 1 paulo None 12 set 19 13:40 col3

Computação Aplicada a Meteorologia

chmod (**change mode**)

As letras “**r**”, “**w**” e “**x**” são as **permissões** (read, write e execute). Além disso, você precisa saber que “**+**” **acrescenta** uma permissão, ao passo que “**-**” **retira**. Se usarmos “**=**”, teremos uma permissão **exata**. Vamos examinar alguns exemplos para podermos entender melhor.

A terminal window titled '~/aulas' with standard window controls (minimize, maximize, close). The prompt is 'paulo@pkubota-PC ~/aulas'. The command '\$ ls -l col3' is entered, and the output is '-rw-r--r-- 1 paulo None 12 set 19 13:40 col3'. The prompt '\$' is shown again.

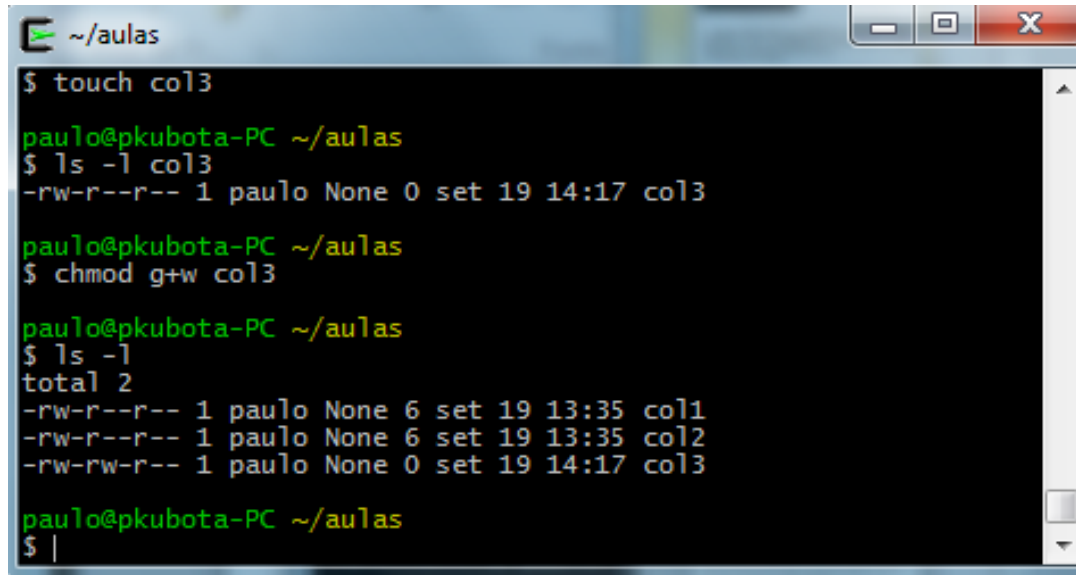
```
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 12 set 19 13:40 col3

paulo@pkubota-PC ~/aulas
$
```

Computação Aplicada a Meteorologia

chmod (**change mode**)

1. Suponhamos que queremos **acrescentar(+)** **permissão** de escrita ao **grupo(g)**. Poderíamos fazer isso da seguinte forma:
\$ chmod g+w exemplo



```
~/aulas
$ touch col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod g+w col3

paulo@pkubota-PC ~/aulas
$ ls -l
total 2
-rw-r--r-- 1 paulo None 6 set 19 13:35 col1
-rw-r--r-- 1 paulo None 6 set 19 13:35 col2
-rw-rw-r-- 1 paulo None 0 set 19 14:17 col3

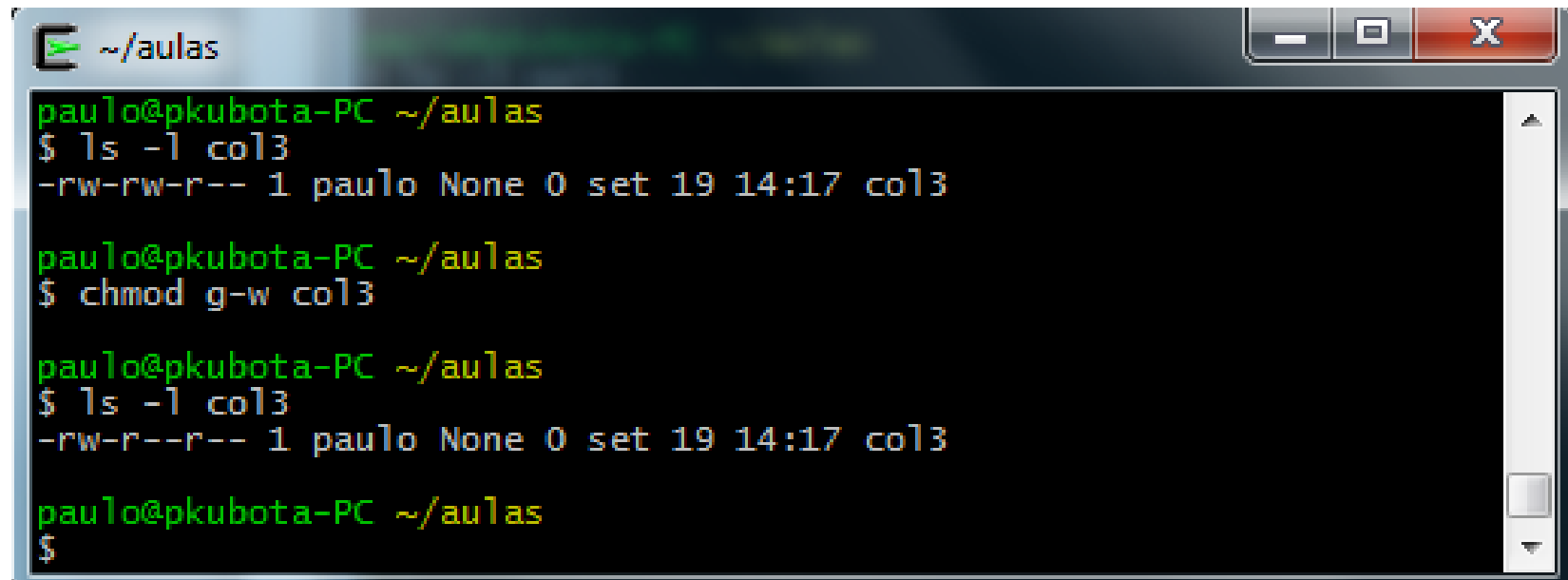
paulo@pkubota-PC ~/aulas
$ |
```


Computação Aplicada a Meteorologia

chmod (**change mode**)

2. Suponhamos agora que acabamos de nos arrepender e queremos tirar a permissão de escrita para o grupo. Poderíamos fazer da seguinte forma:

\$ chmod g-w exemplo



```
~/.aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-rw-r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod g-w col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

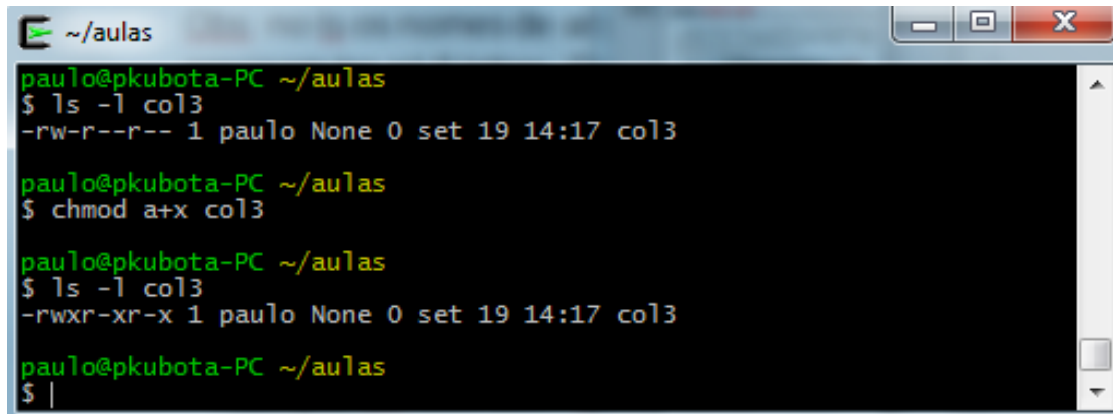
paulo@pkubota-PC ~/aulas
$
```

Computação Aplicada a Meteorologia

chmod (**change mode**)

3. Para acrescentar a permissão de execução a todos os usuários, fazemos:

\$ chmod a+x exemplo



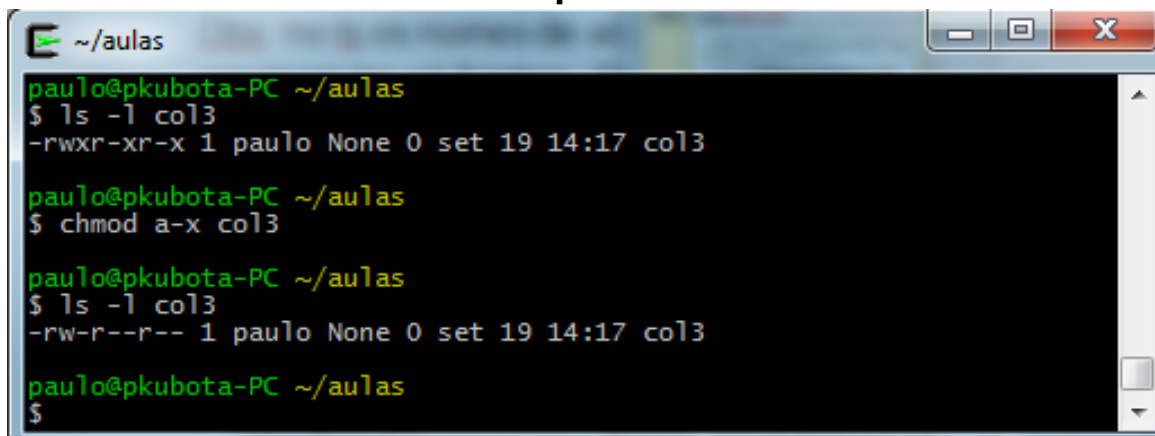
```
~/.aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod a+x col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rwxr-xr-x 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ |
```

\$ chmod a-x exemplo



```
~/.aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rwxr-xr-x 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod a-x col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

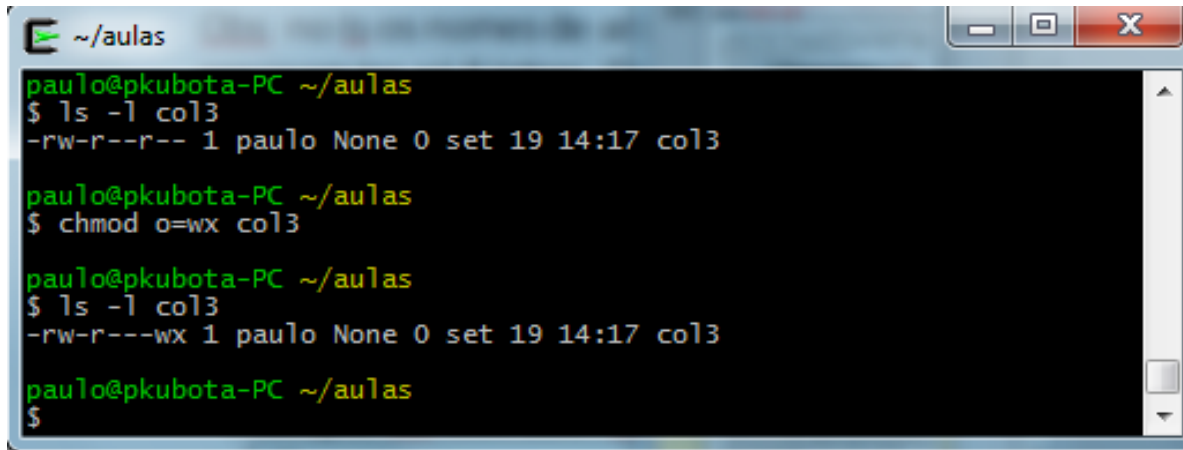
paulo@pkubota-PC ~/aulas
$
```

Computação Aplicada a Meteorologia

chmod (**change mode**)

4. Para que os demais usuários quem sem permissão de leitura, mas tenham permissão de escrita e execução, temos:

\$ chmod o=wx exemplo

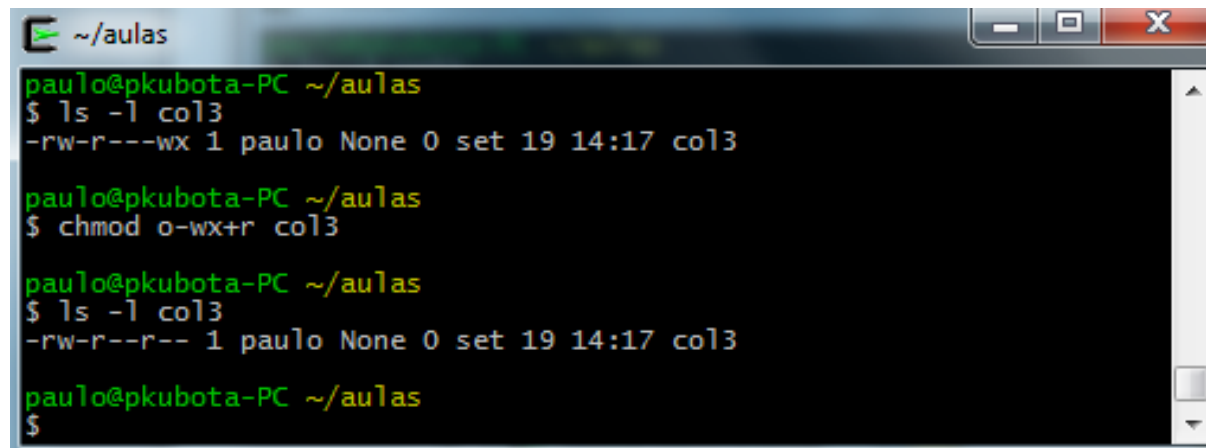


```
~/aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod o=wx col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r---wx 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$
```



```
~/aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r---wx 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod o-wx+r col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$
```

\$ chmod o-wx+r exemplo

Computação Aplicada a Meteorologia

chmod (**change mode**)

O outro modo de **alterar permissões** e usando o chamado **modo octal**. Para usa-lo, e preciso ter em mente o seguinte:

- . **0** - Nenhuma permissão de acesso.
- . **1** - Permissão de execução.
- . **2** - Permissão de escrita.
- . **4** - Permissão de leitura.

A partir disso, podemos obter **qualquer permissão**, **somando os números correspondentes** as permissões desejadas.

- . **3** - Permissão de execução e escrita (1 + 2).
- . **5** - Permissão de execução e leitura (1 + 4).
- . **6** - Permissão de escrita e leitura (2 + 4).
- . **7** - Todas as permissões: execução, escrita e leitura (1 + 2 + 4).

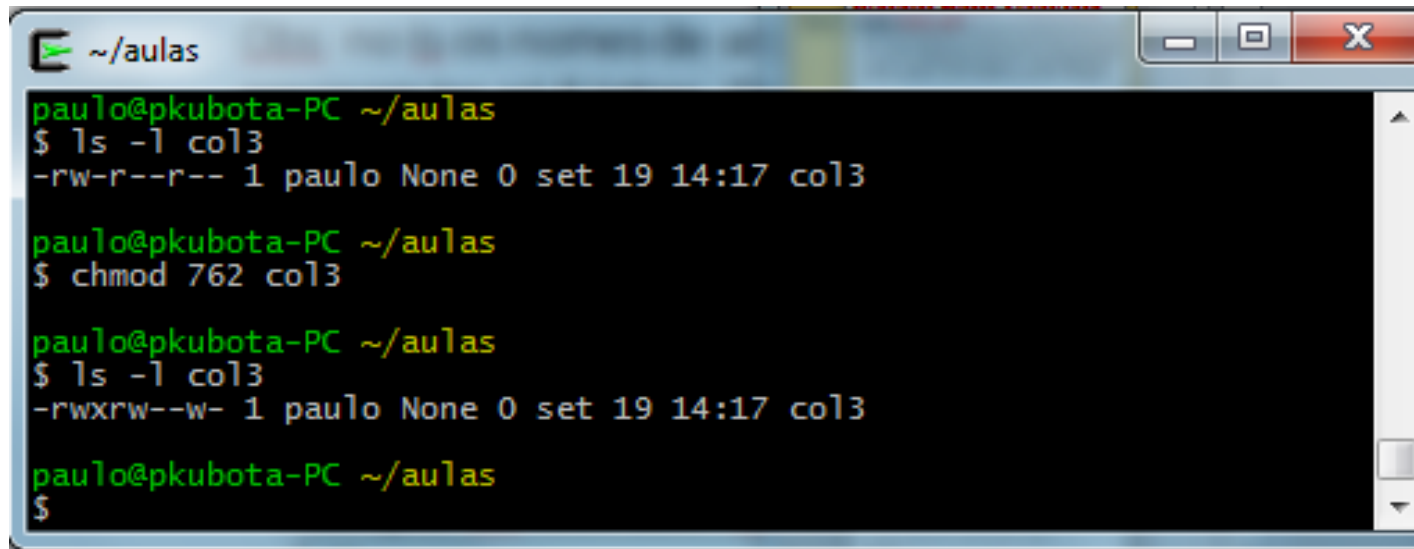
Computação Aplicada a Meteorologia

chmod (change mode)

Com esses algarismos, construímos números com três dígitos (**XYZ**, onde **X** representa a permissão que será definida para o dono, **Y**, a permissão do grupo, e **Z** e a permissão para outros usuários). Vamos mostrar como usar o modo octal.

1. Observe o exemplo a seguir:

\$ chmod **762** exemplo

A terminal window titled '~/aulas' showing a series of commands and their outputs. The user 'paulo' is at 'pkubota-PC'. The first command is 'ls -l col3', showing permissions '-rw-r--r--'. The second command is 'chmod 762 col3'. The third command is 'ls -l col3', showing the updated permissions '-rwxrw--w-'.

```
~/aulas
paulo@pkubota-PC ~/aulas
$ ls -l col3
-rw-r--r-- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$ chmod 762 col3

paulo@pkubota-PC ~/aulas
$ ls -l col3
-rwxrw--w- 1 paulo None 0 set 19 14:17 col3

paulo@pkubota-PC ~/aulas
$
```

-rwxrw--w- 1 paulo None 0 set 19 14:17 col3

Computação Aplicada a Meteorologia

WC

O comando **wc** é usado para **contar linhas**, **palavras** e **bytes** de um **arquivo** ou do que for escrito no terminal.

Sintaxe básica

```
$ wc [opções] [arquivo]
```

Opções

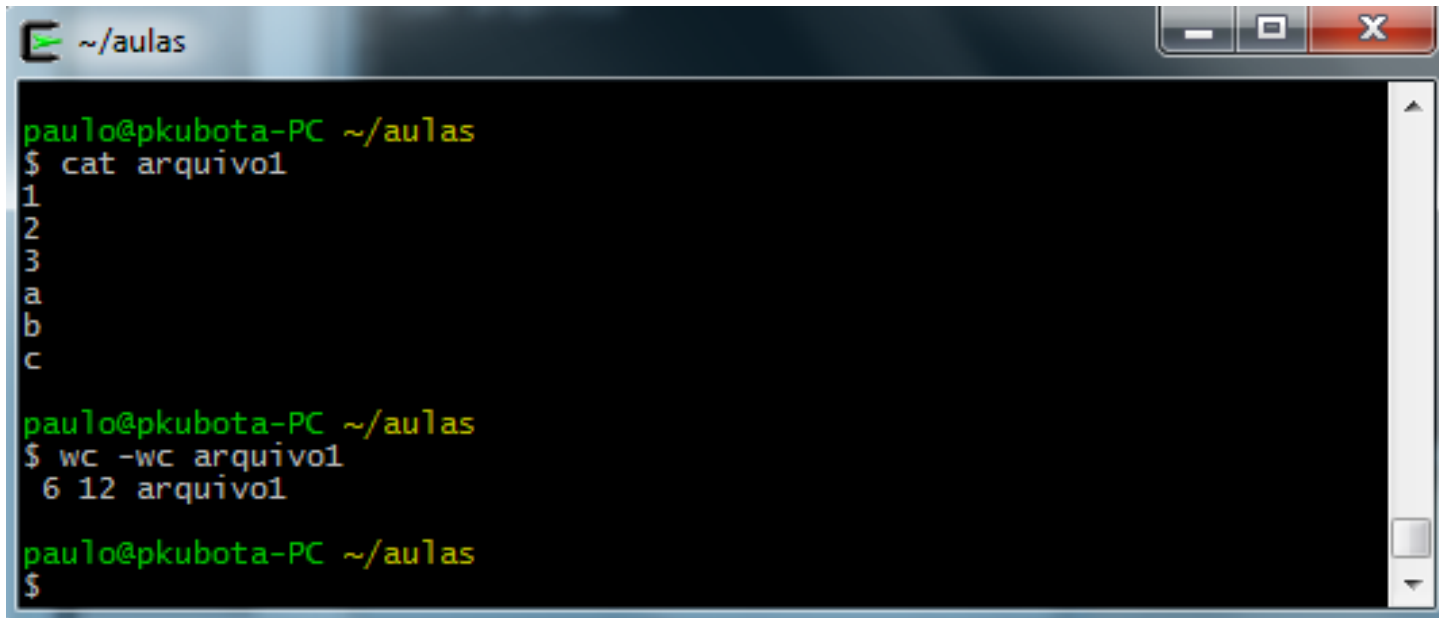
- . **-c**: Imprimir a contagem de bytes.
- . **-l**: Imprimir o número de linhas.
- . **-w**: Imprimir o número de palavras.

Computação Aplicada a Meteorologia

WC

2. Para exibir o numero de palavras e de bytes do arquivo “arquivo2”:

```
$ wc -wc arquivo2  
2 22 arquivo2
```



```
~/aulas  
paulo@pkubota-PC ~/aulas  
$ cat arquivo1  
1  
2  
3  
a  
b  
c  
  
paulo@pkubota-PC ~/aulas  
$ wc -wc arquivo1  
6 12 arquivo1  
  
paulo@pkubota-PC ~/aulas  
$
```

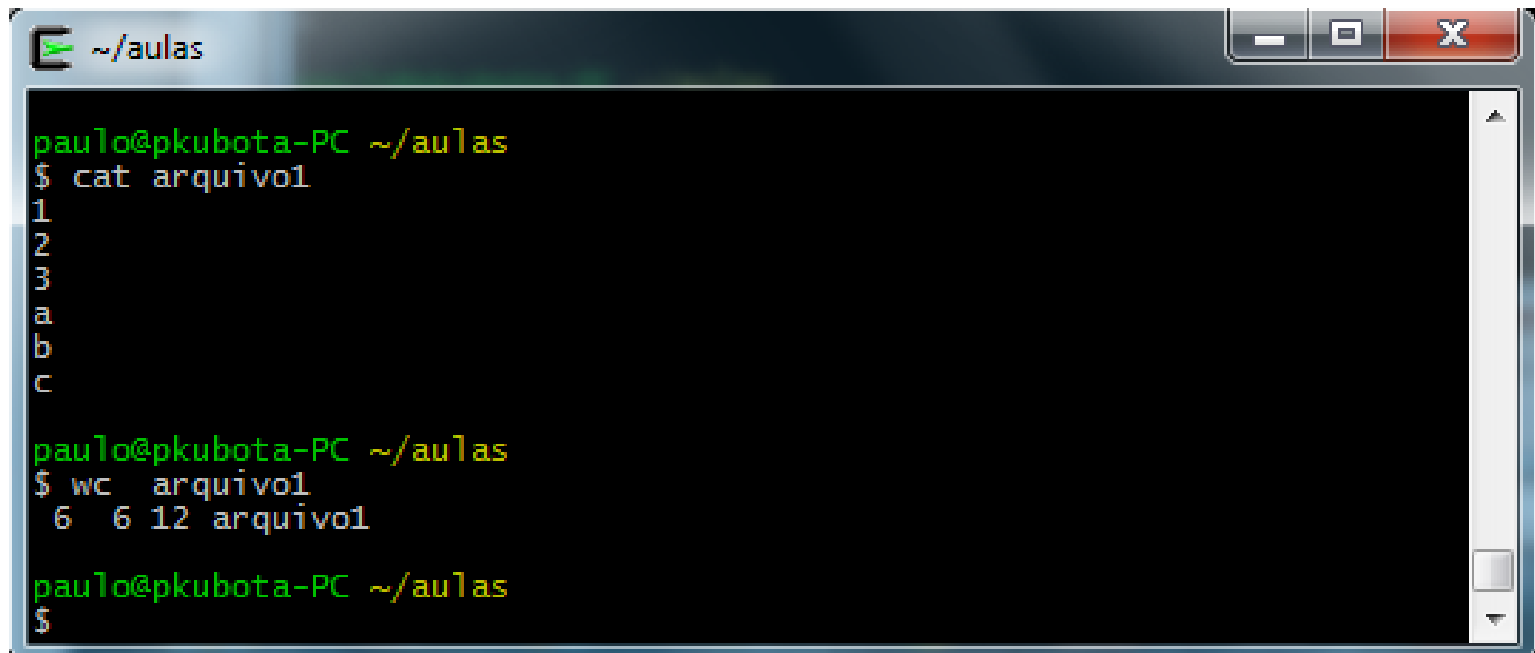
Computação Aplicada a Meteorologia

WC

3. Se usássemos o comando `wc` sem nenhuma opção para “arquivo1”, obteríamos:

```
$ wc arquivo1
```

```
3 3 19 arquivo1
```

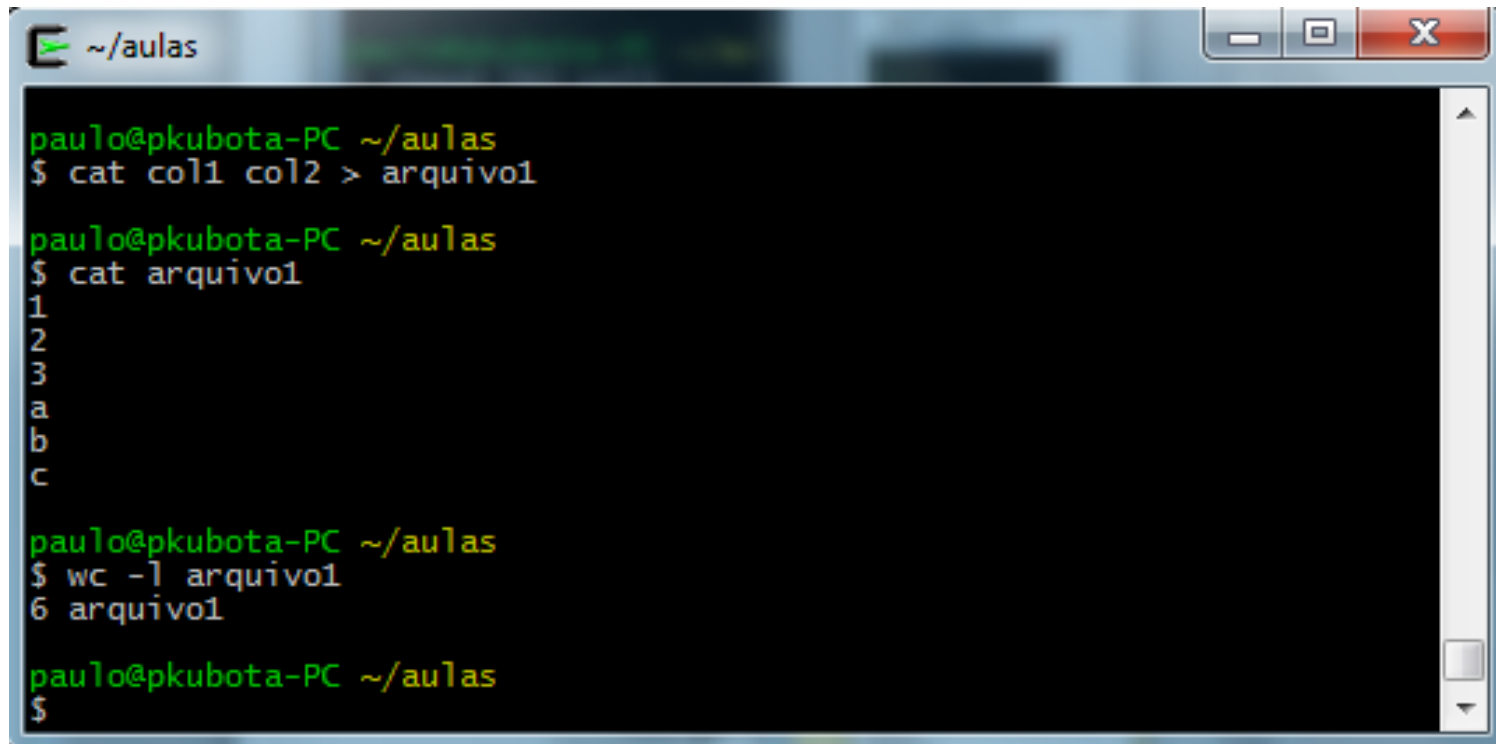


```
~/aulas  
paulo@pkubota-PC ~/aulas  
$ cat arquivo1  
1  
2  
3  
a  
b  
c  
  
paulo@pkubota-PC ~/aulas  
$ wc arquivo1  
6 6 12 arquivo1  
  
paulo@pkubota-PC ~/aulas  
$
```


Computação Aplicada a Meteorologia

1. Para exibir o numero de linhas do arquivo “arquivo1”, usáramos:

```
$ wc -l arquivo1  
3 arquivo1
```

A screenshot of a Linux terminal window titled '~/.aulas'. The window shows a series of commands and their outputs. First, the user runs 'cat col1 col2 > arquivo1'. Then, they run 'cat arquivo1', which outputs the numbers 1, 2, 3 and the letters a, b, c on separate lines. Finally, they run 'wc -l arquivo1', which outputs '6 arquivo1'. The prompt '\$' is visible at the bottom, indicating the command prompt is ready for the next input.

```
paulo@pkubota-PC ~/.aulas  
$ cat col1 col2 > arquivo1  
  
paulo@pkubota-PC ~/.aulas  
$ cat arquivo1  
1  
2  
3  
a  
b  
c  
  
paulo@pkubota-PC ~/.aulas  
$ wc -l arquivo1  
6 arquivo1  
  
paulo@pkubota-PC ~/.aulas  
$
```

Computação Aplicada a Meteorologia

grep (*globally search a regular expression and print*)

O grep realiza **busca** por expressões por linha em **arquivos de texto**.

Exemplos:

```
$ grep ">" genes.fasta
```

Exibe na saída padrão as linhas do arquivo genes.fasta que contém o caractere ">".

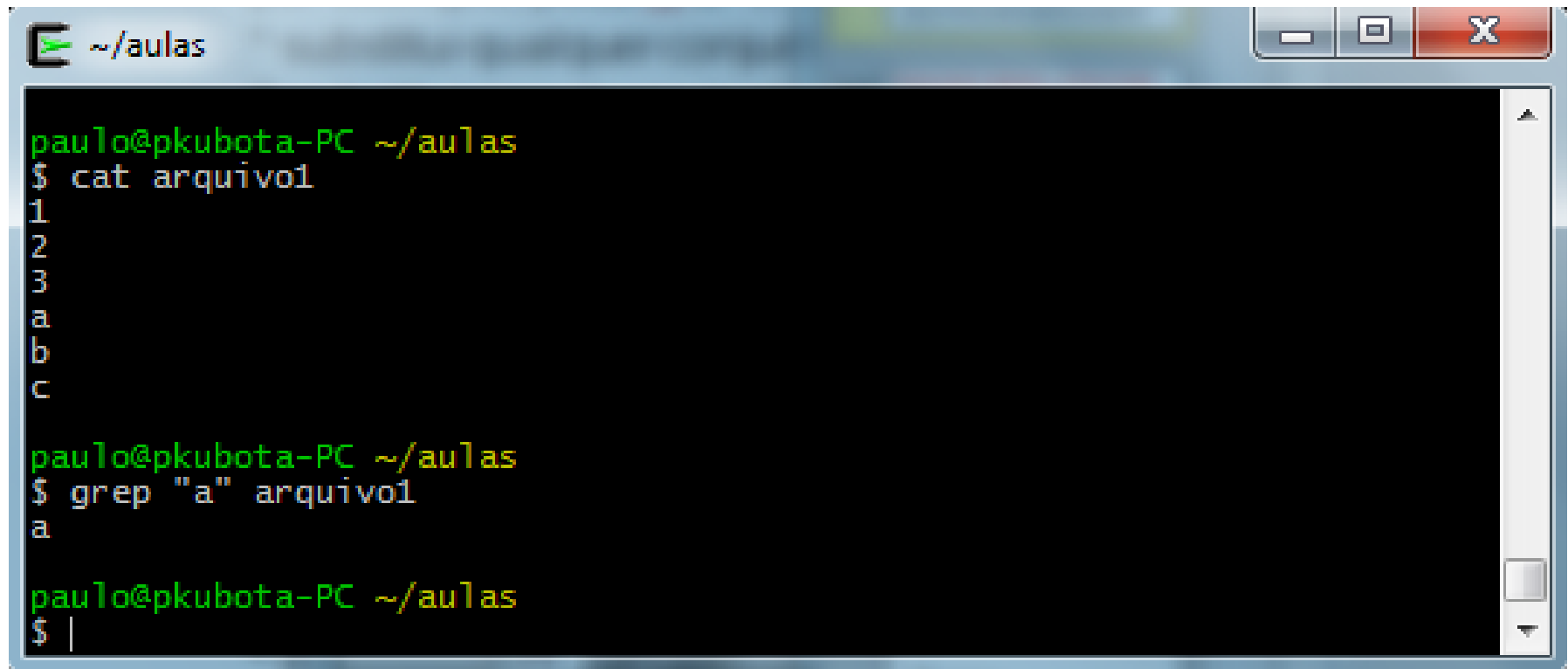
```
$grep -l "GAATTC" *.txt > has_EcoRI.txt
```

Redireciona para o arquivo has_EcoRI.txt uma lista dos arquivos ".txt" do diretório atual que contém "GAATTC".

Computação Aplicada a Meteorologia

grep (*globally search a regular expression and print*)

O grep realiza **busca** por expressões por linha em **arquivos de texto**.

A terminal window titled '~/.aulas' with standard window controls (minimize, maximize, close) in the top right. The terminal shows a user named paulo@pkubota-PC in the directory ~/aulas. The user runs 'cat arquivo1', which outputs lines 1, 2, 3, a, b, and c. Then, the user runs 'grep "a" arquivo1', which outputs only the line 'a'. Finally, the user runs '\$ |', which shows a blank line with a cursor.

```
paulo@pkubota-PC ~/aulas
$ cat arquivo1
1
2
3
a
b
c

paulo@pkubota-PC ~/aulas
$ grep "a" arquivo1
a

paulo@pkubota-PC ~/aulas
$ |
```

Computação Aplicada a Meteorologia

head (output the first part of files)

Head: Exibe na saída padrão as primeiras linhas de um arquivo

```
$ head arquivo.txt
```

Exibe na saída padrão as primeiras 10 linhas de arquivo.txt

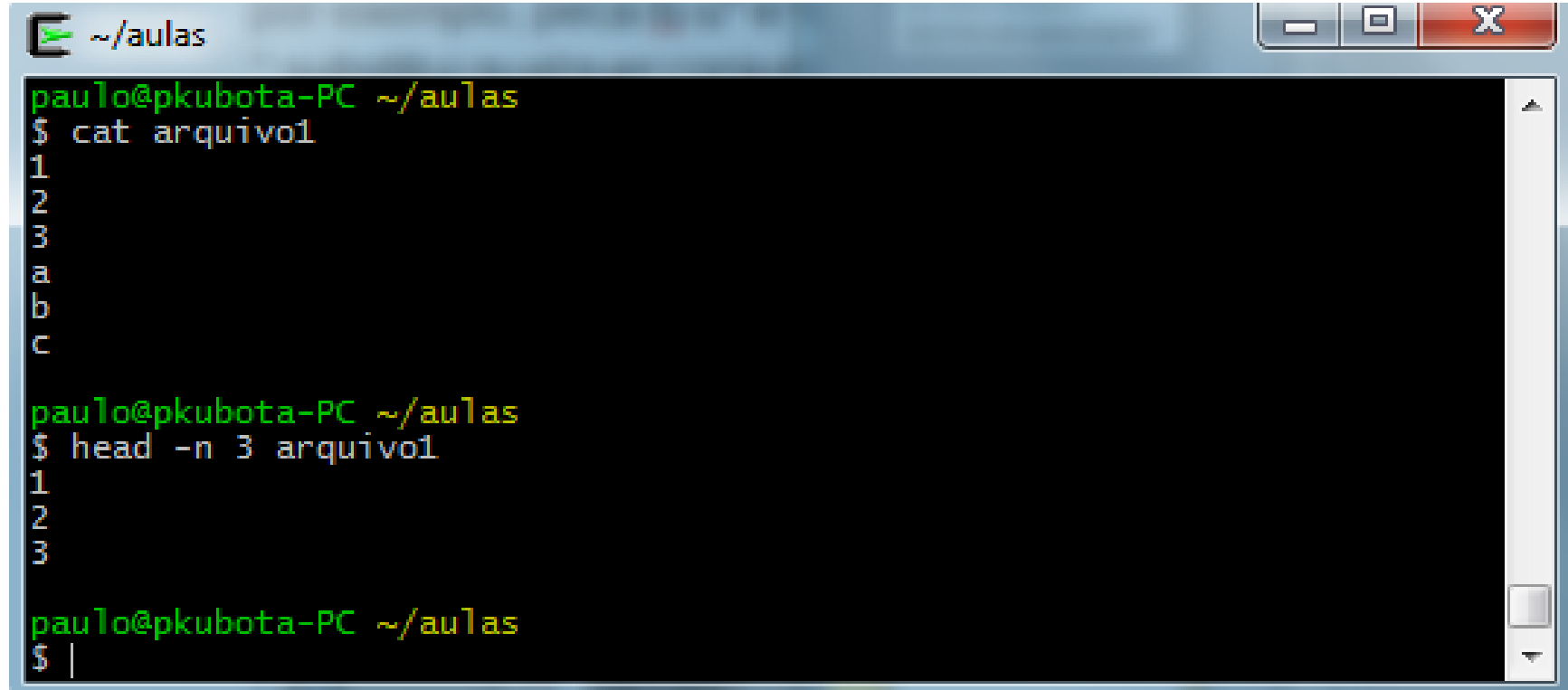
```
$ head -n 21 arquivo.txt
```

Exibe na saída padrão as primeiras 21 linhas de arquivo.txt

Computação Aplicada a Meteorologia

head (output the first part of files)

Head: Exibe na saída padrão as primeiras linhas de um arquivo

A terminal window titled '~/aulas' with standard window controls (minimize, maximize, close) in the top right corner. The terminal shows a user named paulo@pkubota-PC in the directory ~/aulas. The first command executed is 'cat arquivo1', which outputs the contents of the file: '1', '2', '3', 'a', 'b', and 'c' on separate lines. The second command is 'head -n 3 arquivo1', which outputs the first three lines: '1', '2', and '3'. The prompt '\$' is shown at the bottom, indicating the terminal is ready for the next command.

```
~/aulas  
paulo@pkubota-PC ~/aulas  
$ cat arquivo1  
1  
2  
3  
a  
b  
c  
  
paulo@pkubota-PC ~/aulas  
$ head -n 3 arquivo1  
1  
2  
3  
  
paulo@pkubota-PC ~/aulas  
$ |
```

Computação Aplicada a Meteorologia

tail (output the last part of files)

tail: Exibe na saída padrão as primeiras linhas de um arquivo

\$ tail arquivo.txt

Exibe na saída padrão as últimas 10 linhas de arquivo.txt

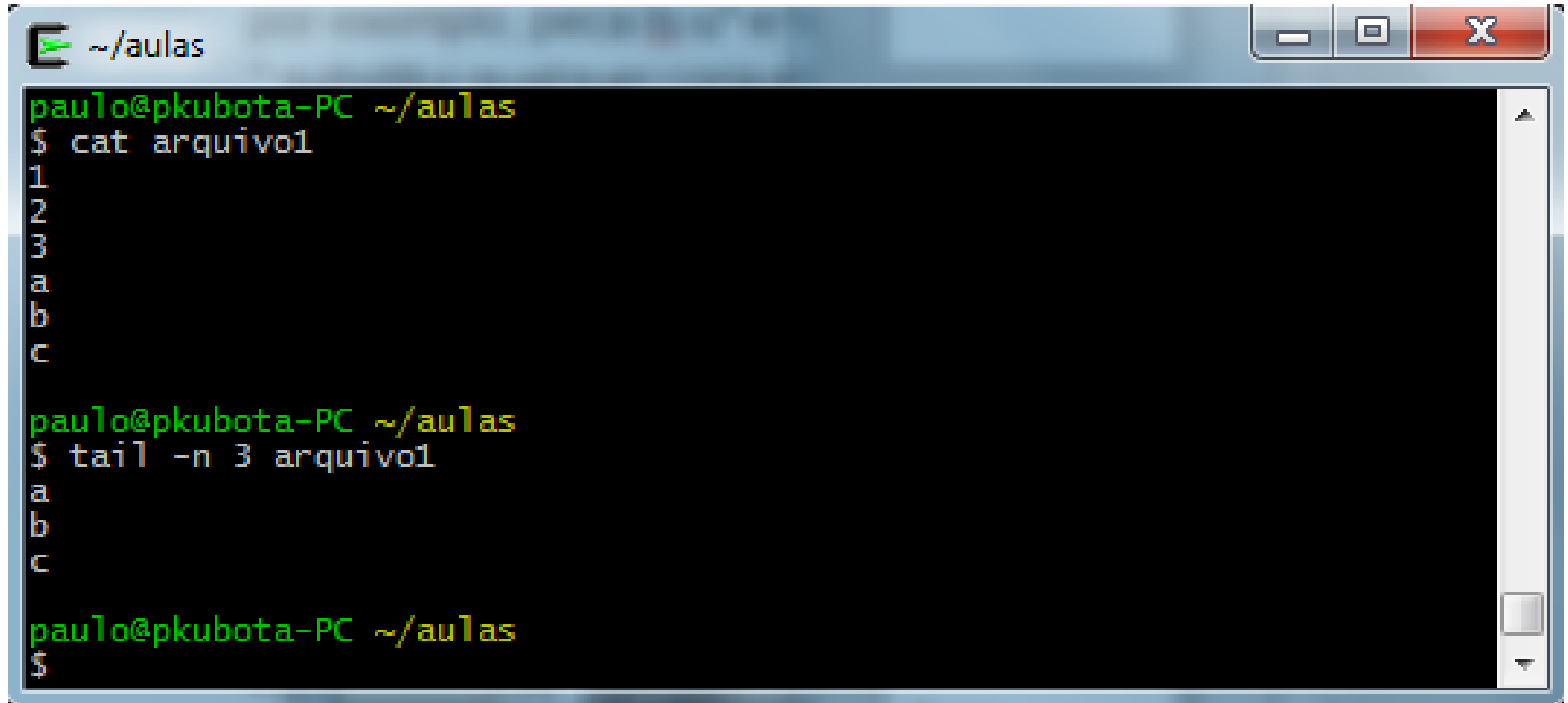
\$ tail -n 21 arquivo.txt

Exibe na saída padrão as últimas 21 linhas de arquivo.txt

Computação Aplicada a Meteorologia

tail (output the last part of files)

tail: Exibe na saída padrão as primeiras linhas de um arquivo

A screenshot of a Linux terminal window titled '~/aulas'. The window shows a user named paulo@pkubota-PC in the directory ~/aulas. The user first runs the command 'cat arquivo1', which outputs the lines '1', '2', '3', 'a', 'b', and 'c'. Then, the user runs 'tail -n 3 arquivo1', which outputs the last three lines: 'a', 'b', and 'c'. The prompt '\$' is shown at the bottom, indicating the command prompt.

```
~/aulas  
paulo@pkubota-PC ~/aulas  
$ cat arquivo1  
1  
2  
3  
a  
b  
c  
  
paulo@pkubota-PC ~/aulas  
$ tail -n 3 arquivo1  
a  
b  
c  
  
paulo@pkubota-PC ~/aulas  
$
```

Computação Aplicada a Meteorologia

sort (sort lines of text files)

O Sort ordena as linha de um arquivo de acordo com parâmetros fornecidos pelo usuário.

Exemplos:

```
$ sort nomes > nomes.sorted
```

Escreve em nomes.sorted o conteúdo de nomes ordenado (tabela ASCII).

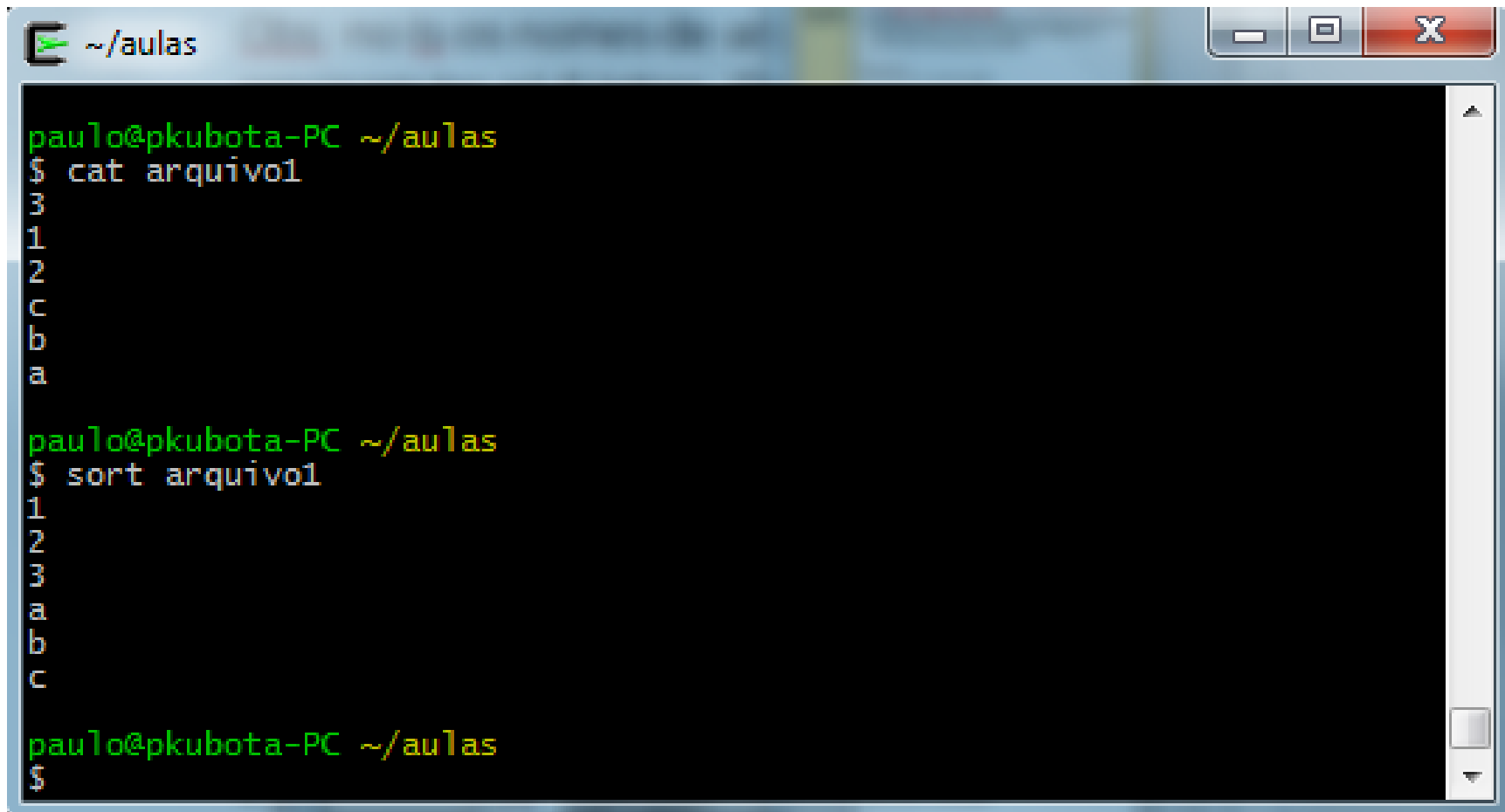
```
$ tail -n +2 arquivo.txt | sort -k 2 -n
```

Retira o cabeçalho da tabela que está no arquivo.txt e ordena as linhas numericamente de acordo com a segunda coluna.

Computação Aplicada a Meteorologia

sort (sort lines of text files)

O Sort ordena as linha de um arquivo de acordo com parâmetros fornecidos pelo usuário.

A terminal window titled '~/.aulas' with standard window controls (minimize, maximize, close) in the top right corner. The prompt is 'paulo@pkubota-PC ~/.aulas'. The first command is '\$ cat arquivo1', followed by the output: '3', '1', '2', 'c', 'b', 'a'. The second command is '\$ sort arquivo1', followed by the output: '1', '2', '3', 'a', 'b', 'c'. The third command is '\$', with no output shown yet.

```
paulo@pkubota-PC ~/.aulas
$ cat arquivo1
3
1
2
c
b
a

paulo@pkubota-PC ~/.aulas
$ sort arquivo1
1
2
3
a
b
c

paulo@pkubota-PC ~/.aulas
$
```

Computação Aplicada a Meteorologia

uniq (remove duplicate lines from a sorted file)

Por padrão, o Uniq remove ocorrências de linhas duplicadas de um arquivo de texto. Para que isso ocorra o arquivo deve estar ordenado.

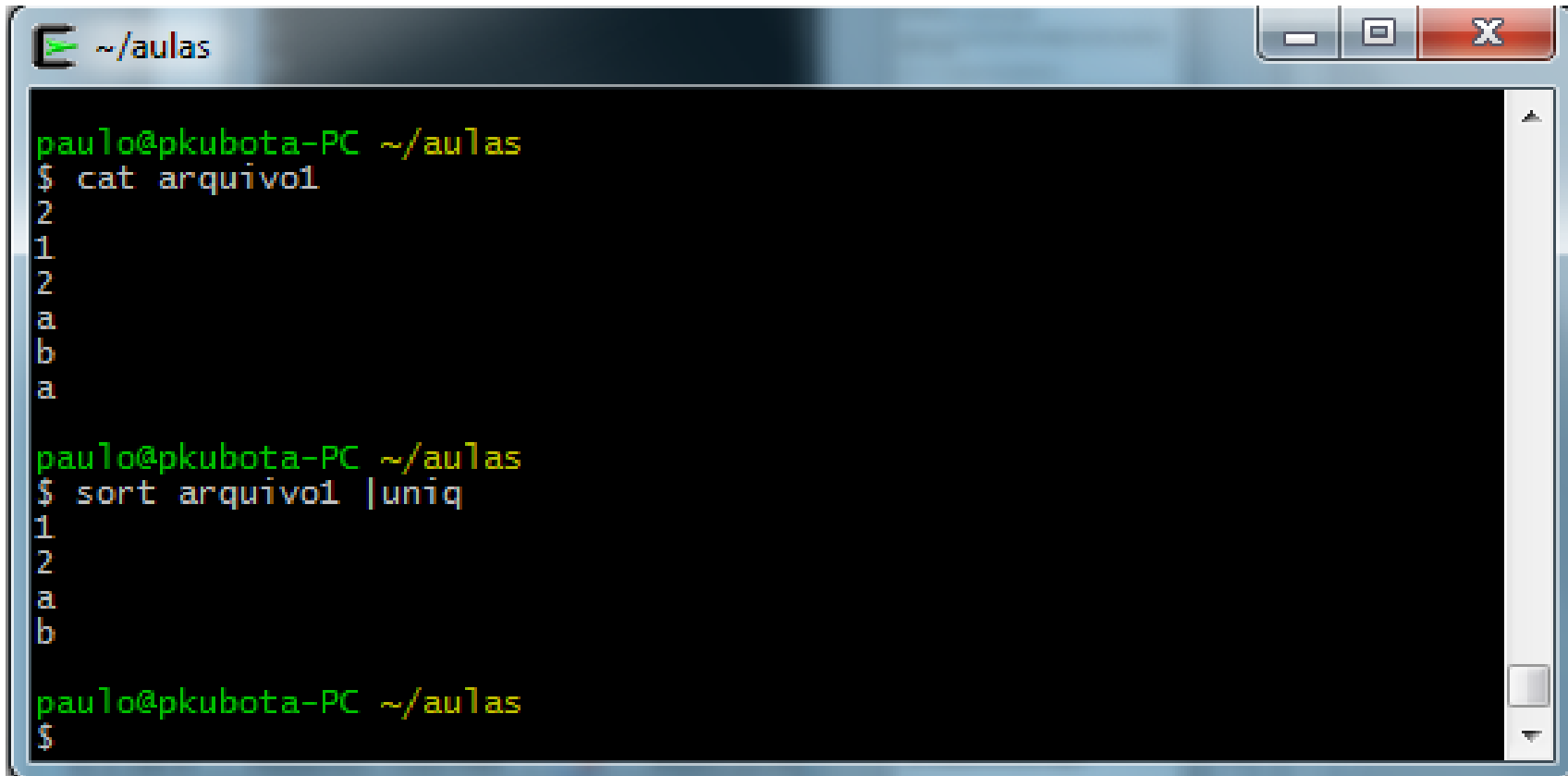
Exemplos: `$ sort nomes | uniq nomes.uniq`

Escreve em `nomes.uniq` o conteúdo removendo os duplicados.

Computação Aplicada a Meteorologia

uniq (remove duplicate lines from a sorted file)

Por padrão, o Uniq remove ocorrências de linhas duplicadas de um arquivo de texto. Para que isso ocorra o arquivo deve estar ordenado.

A terminal window titled '~/aulas' with standard window controls (minimize, maximize, close). The prompt is 'paulo@pkubota-PC ~/aulas'. The first command is '\$ cat arquivo1', which outputs the lines '2', '1', '2', 'a', 'b', 'a'. The second command is '\$ sort arquivo1 | uniq', which outputs the lines '1', '2', 'a', 'b'. The prompt '\$' is shown again at the bottom.

```
paulo@pkubota-PC ~/aulas
$ cat arquivo1
2
1
2
a
b
a

paulo@pkubota-PC ~/aulas
$ sort arquivo1 | uniq
1
2
a
b

paulo@pkubota-PC ~/aulas
$
```

Computação Aplicada a Meteorologia

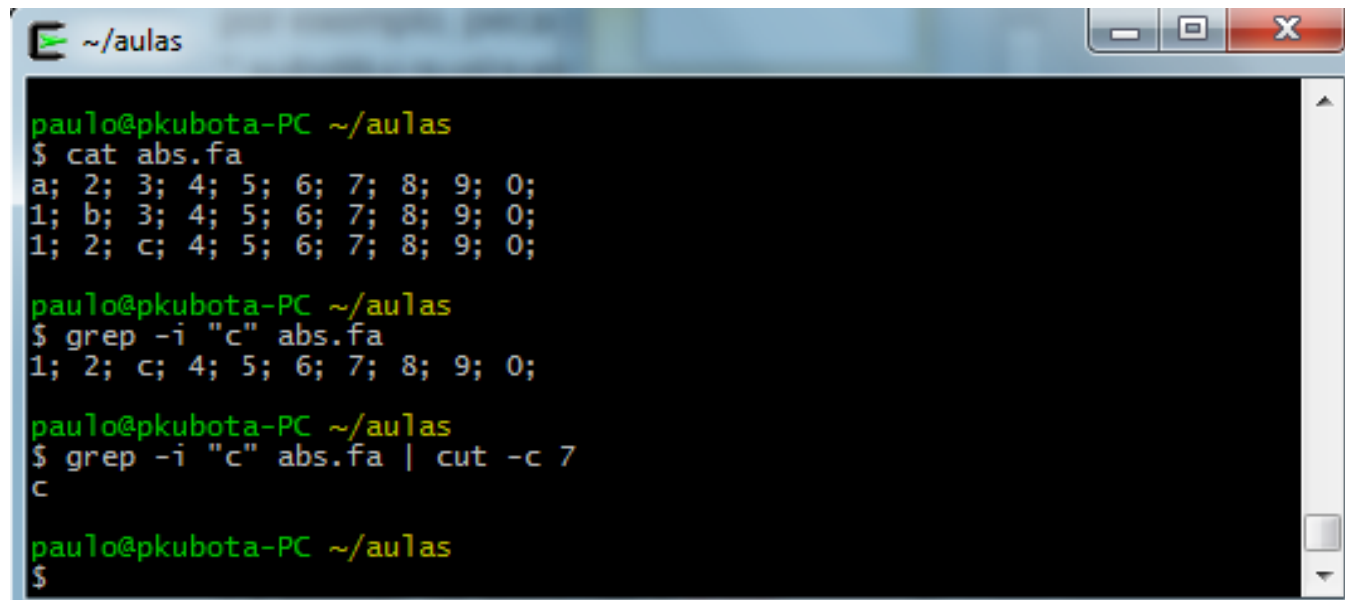
cut (remove sections from each line of files)

O cut fatia o arquivo de entrada verticalmente. É ideal para seleção de colunas de tabelas.

Exemplos:

```
$ grep ">" asb.fa | cut -c 2 > contigs.ids
```

Escreve os nomes dos contigs do arquivo asb.fa em contigs.ids



```
paulo@pkubota-PC ~/aulas
$ cat abs.fa
a; 2; 3; 4; 5; 6; 7; 8; 9; 0;
1; b; 3; 4; 5; 6; 7; 8; 9; 0;
1; 2; c; 4; 5; 6; 7; 8; 9; 0;

paulo@pkubota-PC ~/aulas
$ grep -i "c" abs.fa
1; 2; c; 4; 5; 6; 7; 8; 9; 0;

paulo@pkubota-PC ~/aulas
$ grep -i "c" abs.fa | cut -c 7
c

paulo@pkubota-PC ~/aulas
$
```

Computação Aplicada a Meteorologia

cut (remove sections from each line of files)

O cut fatia o arquivo de entrada verticalmente. É ideal para seleção de colunas de tabelas.

Exemplos:

```
$ grep ">" asb.fa |cut -c 2- > contigs.ids
```

Escreve os nomes dos contigs do arquivo asb.fa em contigs.ids

```
$ cut -f 5,7,9 -d "," ctd.txt > ctd.579.txt
```

Seleciona as colunas 5, 7 e 9 do arquivo ctd.txt, escrevendo-as em ctd.579.txt

Computação Aplicada a Meteorologia

sed

Sed é utilizado para efetuar transformações em texto, como substituições.

Exemplos:

```
$sed s/>Contig/>HybridContig/ assembly.fa >  
assembly.renamed.fa
```

Substitui a primeira ocorrência da string “>Contig” em cada linha do arquivo assembly.fa por “>HybridContig”.

```
$ sed s/t/u/g dna.seq > rna.seq
```

Computação Aplicada a Meteorologia

paste (merge lines of files)

Sintaxe

paste [OPTION]... [FILE].. [FILE2].

Col1 contem

1

2

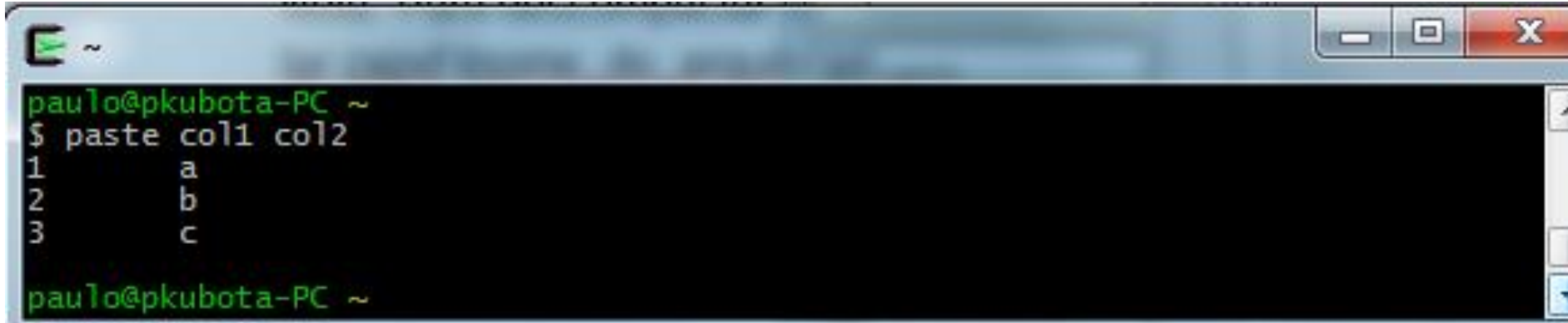
3

Col2 contem

a

b

c

A terminal window with a dark background and green text. The prompt is 'paulo@pkubota-PC ~'. The command 'paste col1 col2' has been entered. The output shows three lines: '1 a', '2 b', and '3 c'. The prompt 'paulo@pkubota-PC ~' is visible again at the bottom.

```
paulo@pkubota-PC ~  
$ paste col1 col2  
1 a  
2 b  
3 c  
  
paulo@pkubota-PC ~
```

Computação Aplicada a Meteorologia

diff (compare files line by line)

SYNOPSIS

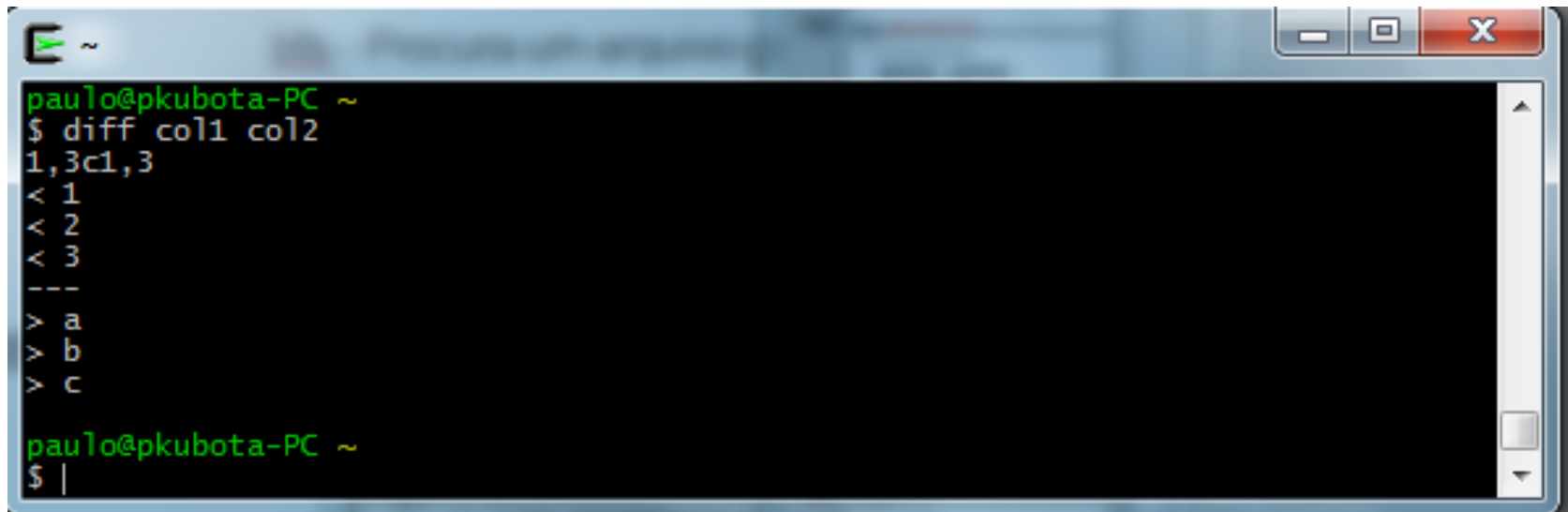
diff [OPTION]... FILES

Col1 contem

1
2
3

Col2 contem

a
b
c



```
paulo@pkubota-PC ~  
$ diff col1 col2  
1,3c1,3  
< 1  
< 2  
< 3  
---  
> a  
> b  
> c  
  
paulo@pkubota-PC ~  
$ |
```


Computação Aplicada a Meteorologia

tar (tape archive)

Sintaxe básica:

```
$ tar [opção] [caminho1/diretorio1]
```

Programa de geração de backup

```
tar -c gera backup
```

```
tar -x restaura backup
```

```
tar -v lista cada arquivo processado
```

```
tar -t lista o conteúdo de um backup
```

Nota: Para descompactar arquivos "tagged"(.tar.gz, .tgz, etc)

```
tar xpvf (nome_do_arquivo)
```

Se o arquivo for "gzipped"(.gz):

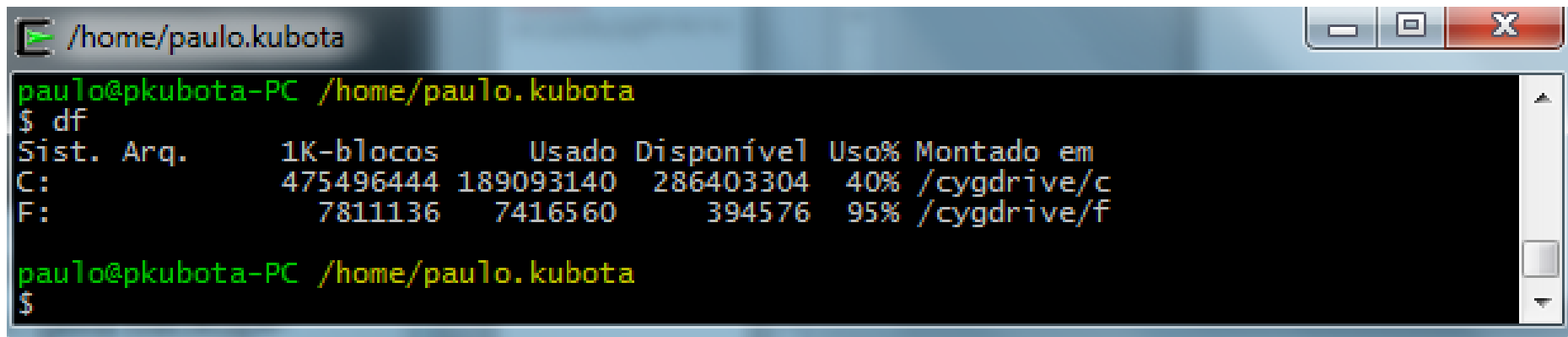
```
gunzip -d (nome_do_arquivo)
```

Computação Aplicada a Meteorologia

df (report file system disk space usage)

Mostra o espaço usado, livre e a capacidade das partições do HD (veja `$man df`)

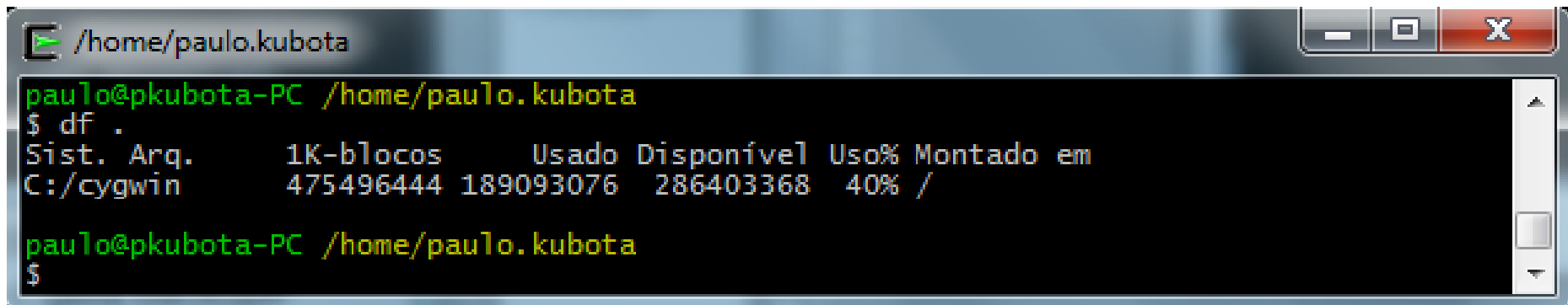
`$df`



```
/home/paulo.kubota
paulo@pkubota-PC /home/paulo.kubota
$ df
Sist. Arq.      1K-blocos      Usado Disponível  Uso% Montado em
C:              475496444 189093140  286403304   40% /cygdrive/c
F:              7811136  7416560   394576    95% /cygdrive/f

paulo@pkubota-PC /home/paulo.kubota
$
```

`$df .`



```
/home/paulo.kubota
paulo@pkubota-PC /home/paulo.kubota
$ df .
Sist. Arq.      1K-blocos      Usado Disponível  Uso% Montado em
C:/cygwin      475496444 189093076  286403368   40% /

paulo@pkubota-PC /home/paulo.kubota
$
```

Computação Aplicada a Meteorologia

(|) Pipe e (>) redirecionamento

| (Pipe)

O pipe (|) é usado para fazer encadeamento de processos, ou seja, faz com que a saída de um comando seja enviada como entrada para o próximo comando.

Observe o exemplo a seguir para entender melhor (o conteúdo de “arquivo1” e “arquivo2” e aquele que foi apresentado junto com o comando cat):

```
$ cat arquivo1 arquivo2 | wc -l
```

5

Computação Aplicada a Meteorologia

(|) Pipe e (>) redirecionamento

> (redirecionamento)

Esta é uma outra forma de direcionar a saída de um comando: diferente do (|), que direcionava a saída de um comando para um outro programa ou comando, o > direciona a saída de um comando para um arquivo ou dispositivo.

Computação Aplicada a Meteorologia

(|) Pipe e (>) redirecionamento

>> (redirecionamento)

O >>, assim como o >, também direciona a saída de um comando para um arquivo, a diferença é que ele não substitui o conteúdo do arquivo, mas acrescenta ao final.

Computação Aplicada a Meteorologia

Sugestões

Finalizamos este modulo com sugestões de sites que podem ajuda-lo a entender melhor o GNU/Linux.

. Rede GNU/Linux: Site da Rede GNU/Linux do Instituto de Matemática e Estatística da USP. www.linux.ime.usp.br

. Make The Move: Tem como objetivo apresentar o Linux e o Software Livre como alternativas viáveis ao sistema em seu computador. www.makethemove.net

. Free Software Foundation: Site da FSF. www.fsf.org

. Debian - The Universal Operating System: Site da distro Debian. www.debian.org

. Ubuntu-BR: Site da Comunidade Ubuntu brasileira. www.ubuntu-br.org

. Fedora Project: Site da distro Fedora. www.fedoraproject.org

. Distro Watch: Notícias e informações sobre distribuições Linux e BSD. distrowatch.com

. Google: Dispensa apresentações. www.google.com

Computação Aplicada a Meteorologia

Referências Bibliográficas

[2] Ficheiro: Fluxbox.png - Wikipedia, a enciclopedia livre, disponvel in <http://pt.wikipedia.org/wiki/Ficheiro:Fluxbox.png> [Janeiro de 2010]

[3] Ficheiro: KDE 4.png - Wikipedia, a enciclopedia livre, disponvel in http://pt.wikipedia.org/wiki/Ficheiro:KDE_4.png [Janeiro de 2010]

[4] Ficheiro: Linus Torvalds.jpeg - Wikipedia, a enciclopedia livre, disponvel in http://pt.wikipedia.org/wiki/Ficheiro:Linus_Torvalds.jpeg [Janeiro de 2010]

[5] Ficheiro: Richard Matthew Stallman.jpeg - Wikipedia, a enciclopedia livre, disponvel in http://pt.wikipedia.org/wiki/Ficheiro:Richard_Matthew_Stallman.jpeg [Janeiro de 2010]

[6] Filesystem Hierarchy Standard - Wikipedia, a enciclopedia livre, disponvel in http://pt.wikipedia.org/wiki/Filesystem_Hierarchy_Standard [Fevereiro de 2010]

[7] GNU General Public License - Wikipedia, a enciclopedia livre, disponvel in http://pt.wikipedia.org/wiki/GNU_General_Public_License [Janeiro de 2010]