



SHELL SCRIPT

na prática

Edson Araújo

- Doutorando em Biotecnologia
- Mestre em Modelagem Molecular
- Bacharel em Sistemas de Informação
- Técnico em Redes de Computadores
- Webmaster do Laboratório CASTA
- Usuário Linux desde 2009
- Aprendeu Shell Script em 2010



SHELL SCRIPT

na prática

Metodologia

Teoria

Exemplos

Prática

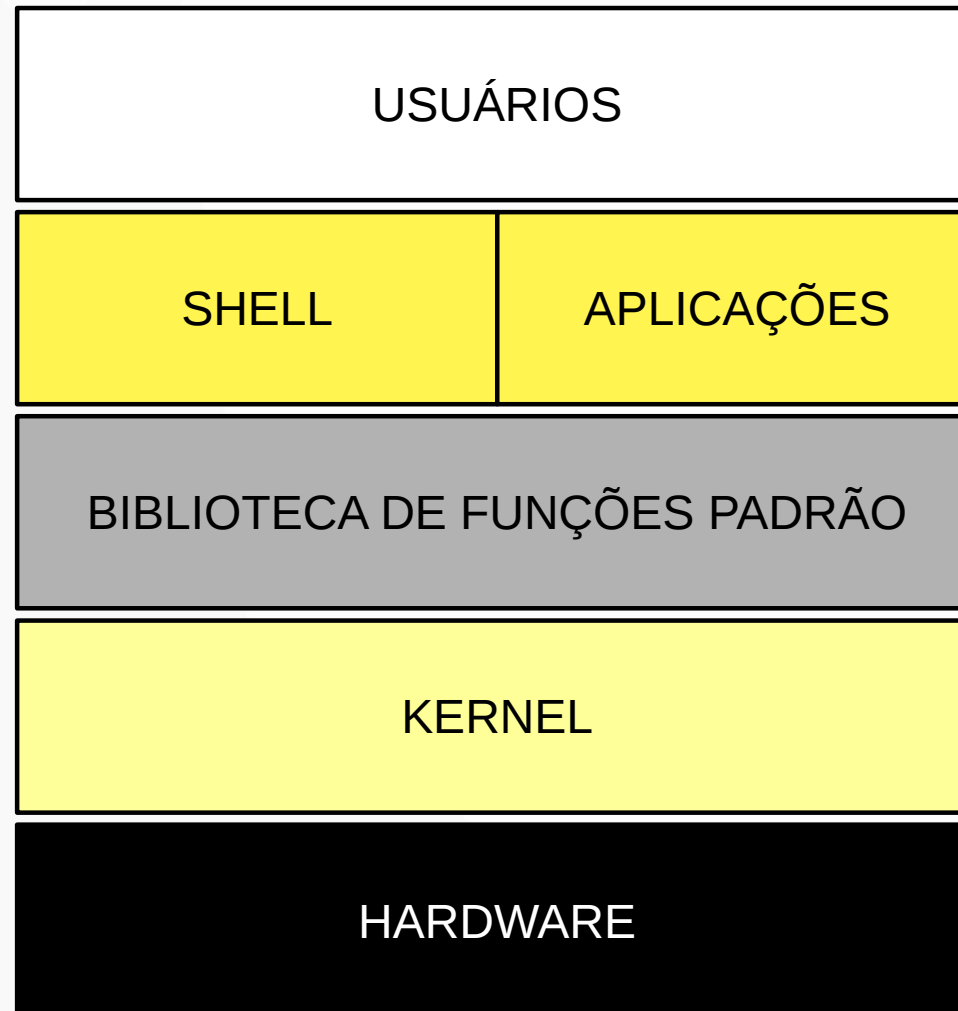
Resumo

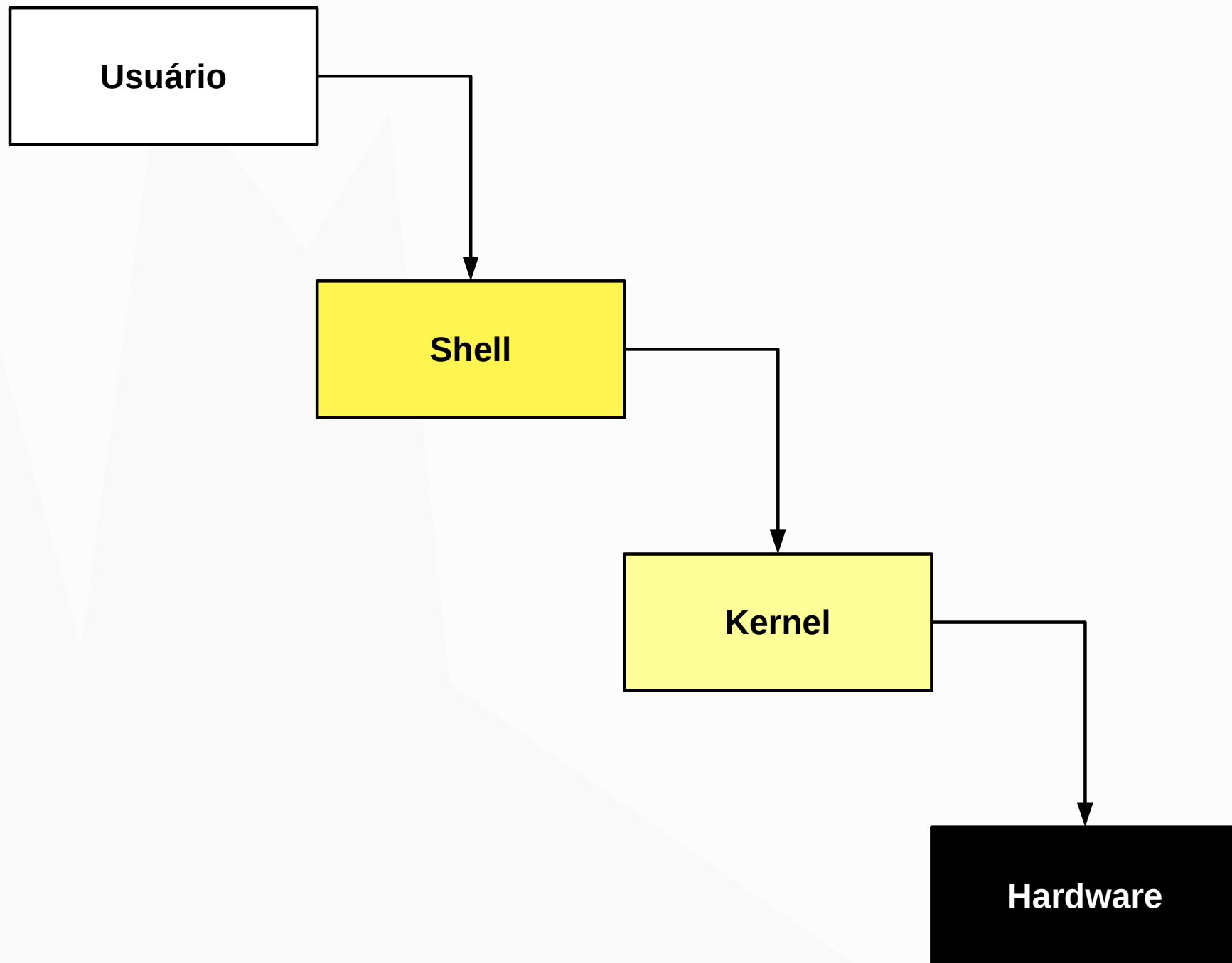
- Arquivo Shell Script
- Variáveis
- If, for e while
- Interação com usuário
- Comando test
- Cores
- Parâmetros
- Opções por linha de comando
- Dialog

O que é Shell?

- **Prompt** da linha de comando do Unix e do Linux
- Acesso: **Terminal**
- Cada comando digitado é lido, verificado e enviado ao sistema operacional para ser de fato executado
- A **ponte entre o usuário e o Kernel**

Arquitetura Linux





Tipos de Shell

- Csh ou C Shell
- Bourne Shell ou sh
- Bourne-Again Shell ou bash
- Korn Shell ou Ksh

Interpretador: Bash

O que é Shell Script ?

- Script é um **arquivo que guarda vários comandos do sistema e que pode ser executado pelo usuário**
- Cabeçalho do arquivo:
`#!/bin/bash`

Exemplo 00 - Conectando Wifi

```
#!/bin/bash
```

```
ifconfig wlp9s0 up
```

```
iwconfig wlp9s0 essid Linux
```

```
dhclient wlp9s0
```

Variáveis

```
$ OFICINA="Oficina de Shell Script"
```

```
$ HOJE=$(date)
```

```
$ echo "$OFICINA em: $HOJE"
```

```
$ echo $HOJE
```

```
$ unset HOJE
```

```
$ echo $HOJE
```

Variáveis (Globais)

```
$ echo $USER
```

```
$ echo $HOME
```

```
$ echo $PS1 "mensagem"
```

```
$ echo $GDMSESSION
```

Lista completa: `$ env`

If, for e while

```
if COMANDO
then
    comandos
else
    comandos
fi
```

```
for VAR in LIST
do
    comandos
done
```

```
while COMANDO
do
    comandos
done
```

Primeiro Shell Script

- Editor de texto: **Nano**, Vim, Gedit, etc
- Nome do arquivo: extensão **.sh**
- Chamada do Shell na primeira linha: **#!/bin/bash**
- Tornar o arquivo executável: **sudo chmod +x arquivo.sh**
- Executar script: **\$./arquivo.sh**

Exemplo 01 - Capturando informações do sistema

```
#!/bin/bash
```

```
echo "Data e Horário:"
```

```
date
```

```
echo
```

```
echo "Uso do disco:"
```

```
df
```

```
echo
```

```
echo "Nome do usuário:"
```

```
echo $USER
```

Exemplo 01 - Capturando informações do sistema

```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help
[earaujo@localhost oficina]$ ./exemplo01.sh
Data e Horário:
Thu May 10 21:09:35 -03 2018

Uso do disco:

```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
devtmpfs	1958136	0	1958136	0%	/dev
tmpfs	1972068	32452	1939616	2%	/dev/shm
tmpfs	1972068	1304	1970764	1%	/run
tmpfs	1972068	0	1972068	0%	/sys/fs/cgroup
/dev/mapper/fedora_localhost--live-root	51343840	6209008	42497008	13%	/
tmpfs	1972068	1608	1970460	1%	/tmp
/dev/sda6	999320	224152	706356	25%	/boot
/dev/mapper/fedora_localhost--live-home	172040424	5490440	157741128	4%	/home
tmpfs	394412	12	394400	1%	/run/user/1000

```
Nome do usuário:
earaujo
[earaujo@localhost oficina]$
```

Exemplo 02 - Interagindo com usuário

```
#!/bin/bash
```

```
echo "Buscar informações  
do sistema ? [s / n] "
```

```
read RESPOSTA
```

```
test "$RESPOSTA" = "n" &&  
exit
```

```
echo "Data e Horário:"
```

```
date
```

```
echo
```

```
echo "Uso do disco:"
```

```
df
```

```
echo
```

```
echo "Nome do usuário:"
```

```
echo $USER
```

Exemplo 02 - Interagindo com usuário

```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help

[earaujo@localhost oficina]$ ./exemplo02.sh
Deseja buscar informações do sistema? [s / n]
n
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo02.sh
Deseja buscar informações do sistema? [s / n]
s
Data e Horário:
Thu May 10 21:15:02 -03 2018

Uso do disco:

```

Filesystem	1K-blocks	Used	Available	Use%	Mounted
on					
devtmpfs	1958136	0	1958136	0%	/dev
tmpfs	1972068	32452	1939616	2%	/dev/shm
tmpfs	1972068	1304	1970764	1%	/run
tmpfs	1972068	0	1972068	0%	/sys/fs
/cgroup					
/dev/mapper/fedora_localhost--live-root	51343840	6209008	42497008	13%	/
tmpfs	1972068	1876	1970192	1%	/tmp
/dev/sda6	999320	224152	706356	25%	/boot
/dev/mapper/fedora_localhost--live-home	172040424	5490548	157741020	4%	/home
tmpfs	394412	16	394396	1%	/run/usb

Exemplo 03 - Atualizando o sistema

```
#!/bin/bash
```

```
echo "Deseja atualizar o  
sistema? [s / n]"
```

```
read RESPOSTA
```

```
if test "$RESPOSTA" = "n" -o  
"$RESPOSTA" = "N"
```

```
then
```

```
    exit
```

```
elif test "$RESPOSTA" = "s"  
-o "$RESPOSTA" = "S"
```

```
then
```

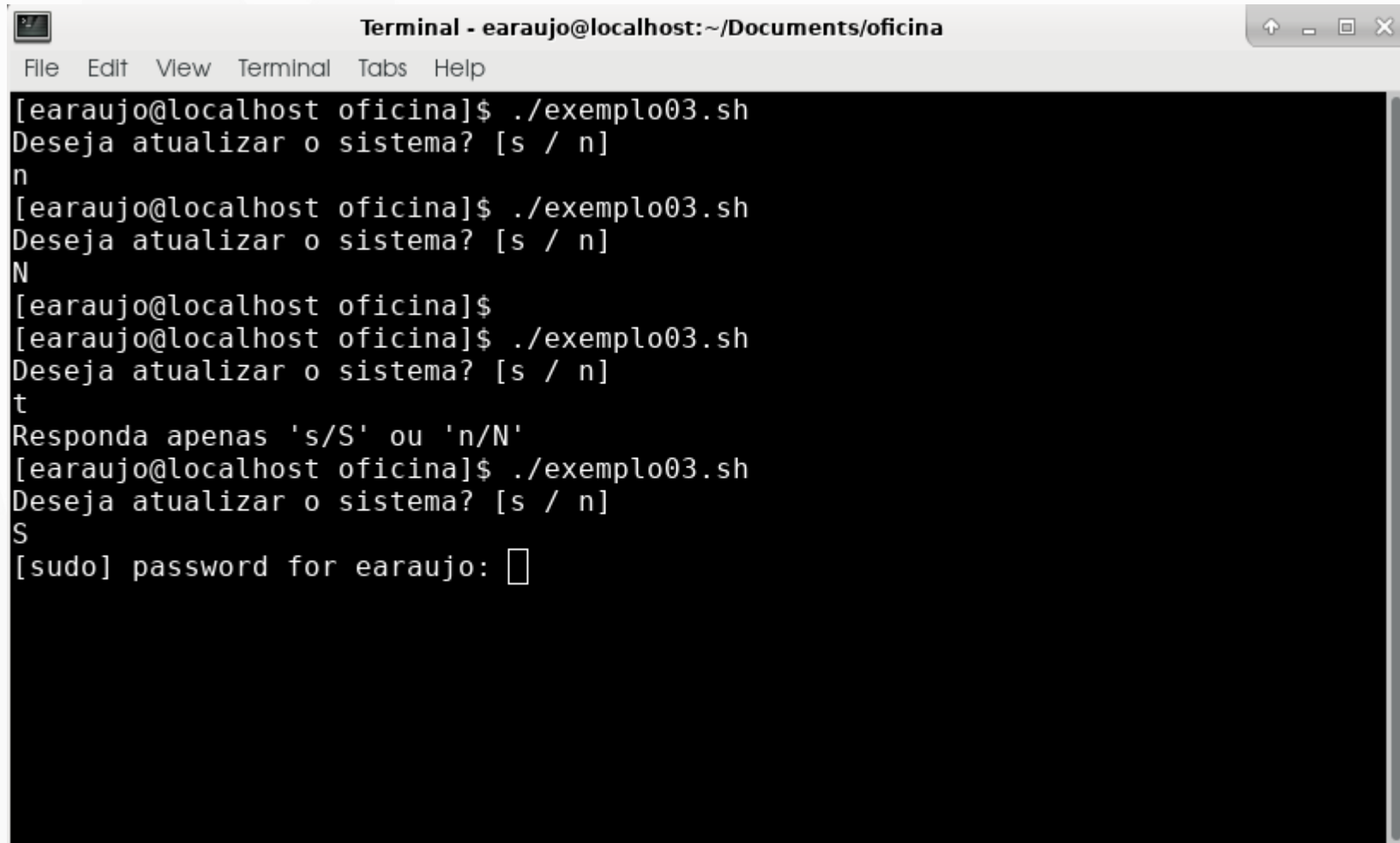
```
    sudo dnf update -y
```

```
else
```

```
echo "Responda apenas  
's/S' ou 'n/N' "
```

```
fi
```

Exemplo 03 - Atualizando o sistema



```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help
[earaujo@localhost oficina]$ ./exemplo03.sh
Deseja atualizar o sistema? [s / n]
n
[earaujo@localhost oficina]$ ./exemplo03.sh
Deseja atualizar o sistema? [s / n]
N
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo03.sh
Deseja atualizar o sistema? [s / n]
t
Responda apenas 's/S' ou 'n/N'
[earaujo@localhost oficina]$ ./exemplo03.sh
Deseja atualizar o sistema? [s / n]
S
[sudo] password for earaujo: 
```

Comando test

- Faz vários tipos de testes em números, textos e arquivos

Exemplo 04 - Testando arquivos

```
#!/bin/bash
```

```
echo -n "Digite o nome do arquivo: "
```

```
read ARQUIVO
```

```
test -e "$ARQUIVO" && echo "$ARQUIVO existe"
```

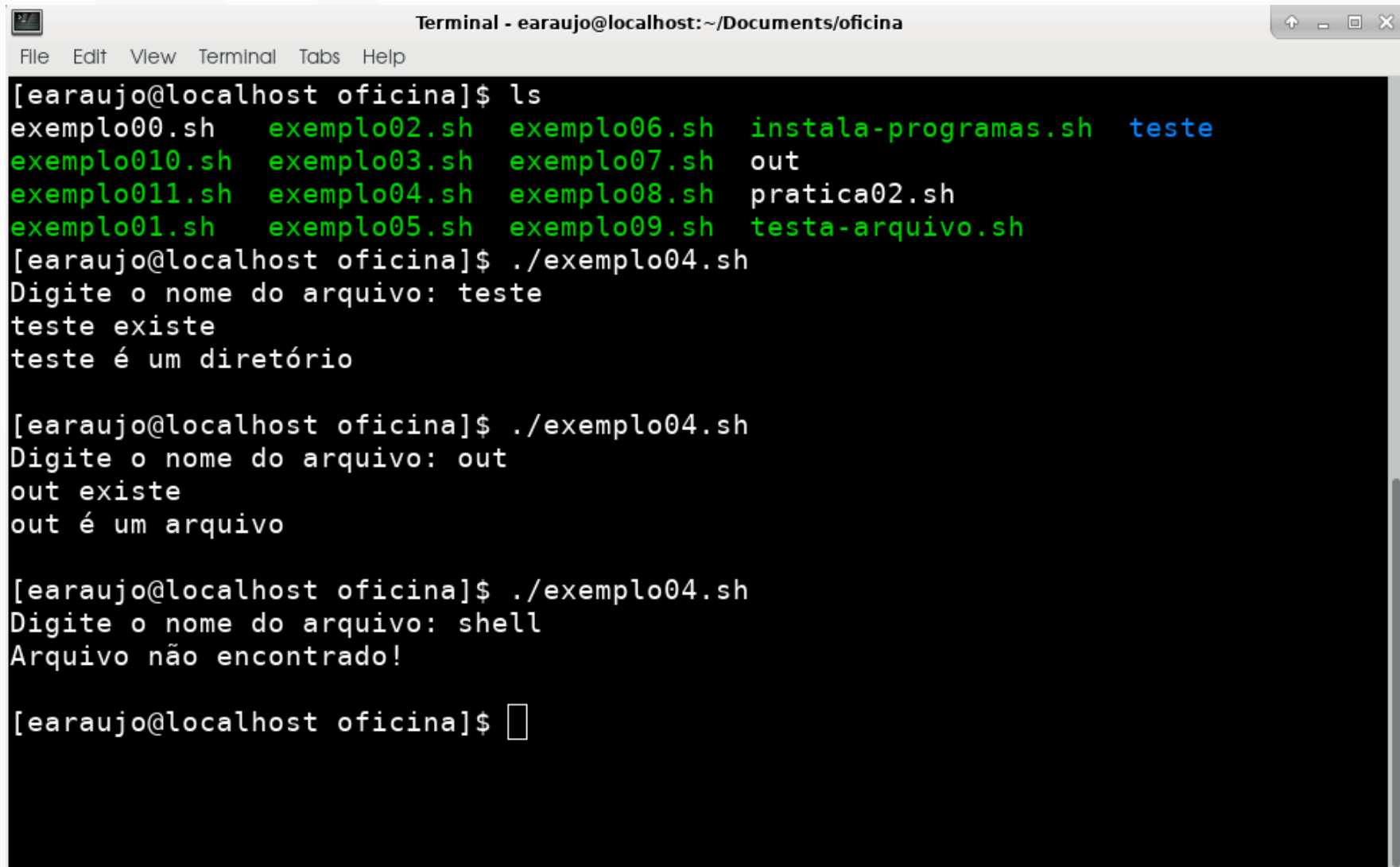
```
test -d "$ARQUIVO" && echo "$ARQUIVO é um diretório"
```

```
test -f "$ARQUIVO" && echo "$ARQUIVO é um arquivo"
```

```
test -f "$ARQUIVO" -o -d "$ARQUIVO" || echo "Arquivo não encontrado"
```

```
echo
```


Exemplo 04 - Testando arquivos



```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help

[earaujo@localhost oficina]$ ls
exemplo00.sh  exemplo02.sh  exemplo06.sh  instala-programas.sh  teste
exemplo010.sh exemplo03.sh  exemplo07.sh  out
exemplo011.sh exemplo04.sh  exemplo08.sh  pratica02.sh
exemplo01.sh  exemplo05.sh  exemplo09.sh  testa-arquivo.sh

[earaujo@localhost oficina]$ ./exemplo04.sh
Digite o nome do arquivo: teste
teste existe
teste é um diretório

[earaujo@localhost oficina]$ ./exemplo04.sh
Digite o nome do arquivo: out
out existe
out é um arquivo

[earaujo@localhost oficina]$ ./exemplo04.sh
Digite o nome do arquivo: shell
Arquivo não encontrado!

[earaujo@localhost oficina]$
```

Cores na tela

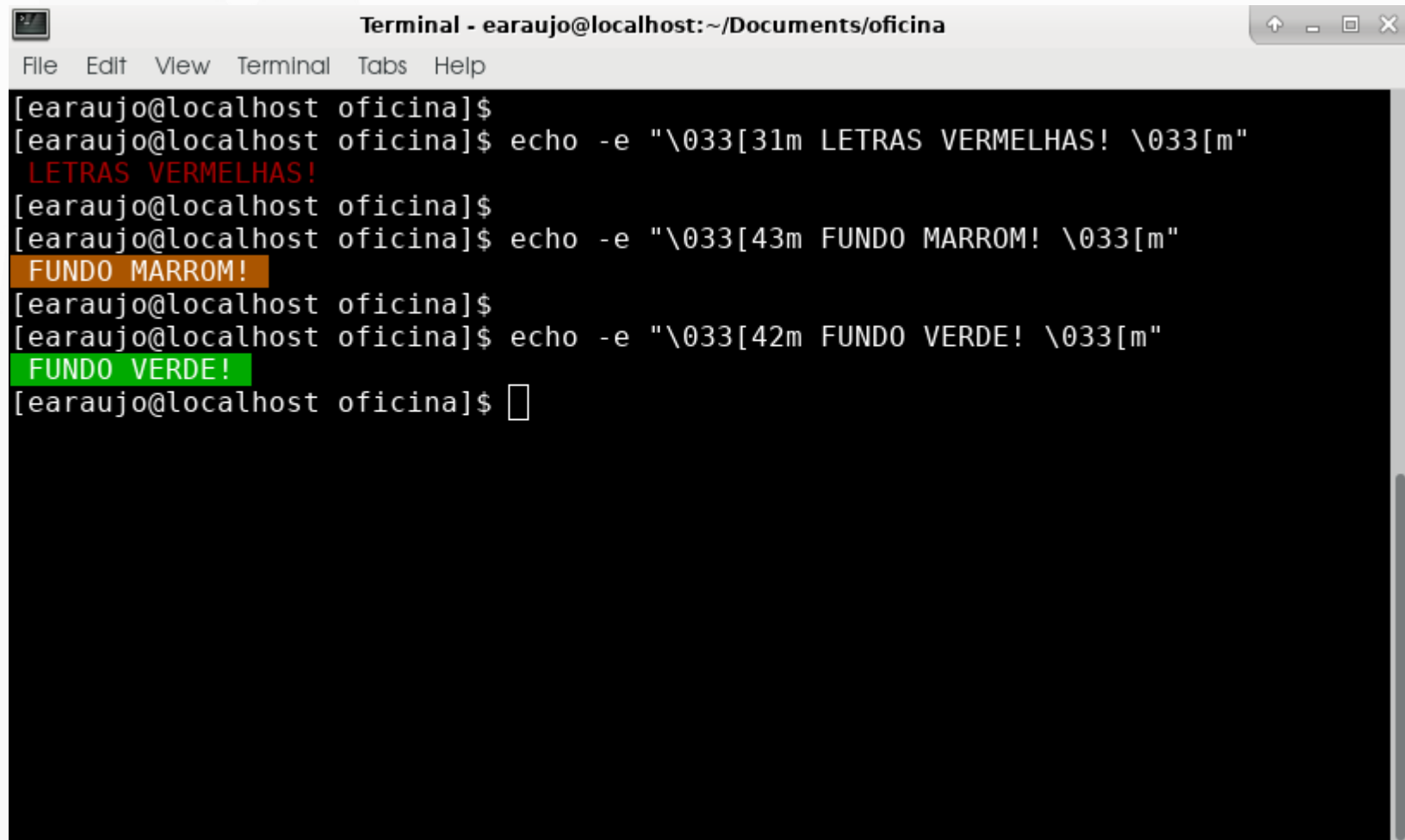
- Sintaxe: (ESC[m) \033[n1; n2; ...m

```
$ echo -e "\033[31m LETRAS VERMELHAS! \033[m"
```

```
$ echo -e "\033[43m FUNDO MARROM! \033[m"
```

```
$ echo -e "\033[42m FUNDO VERDE! \033[m"
```

Cores na tela

A terminal window titled "Terminal - earaujo@localhost: ~/Documents/oficina" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal shows three commands using 'echo -e' to print text with ANSI color codes. The first command prints "LETRAS VERMELHAS!" in red. The second command prints "FUNDO MARROM!" on a brown background. The third command prints "FUNDO VERDE!" on a green background. The prompt is "[earaujo@localhost oficina]\$".

```
[earaujo@localhost oficina]$  
[earaujo@localhost oficina]$ echo -e "\033[31m LETRAS VERMELHAS! \033[m"  
LETRAS VERMELHAS!  
[earaujo@localhost oficina]$  
[earaujo@localhost oficina]$ echo -e "\033[43m FUNDO MARROM! \033[m"  
FUNDO MARROM!  
[earaujo@localhost oficina]$  
[earaujo@localhost oficina]$ echo -e "\033[42m FUNDO VERDE! \033[m"  
FUNDO VERDE!  
[earaujo@localhost oficina]$
```

Tabela de cores

Código	Descrição
0	Texto normal, sem cores
1	Cor brilhante
30	Texto preto (ou cinza)
31	Texto vermelho
32	Texto verde
33	Texto marrom (ou amarelo)
34	Texto azul
35	Texto roxo
36	Texto ciano
37	Texto cinza (ou branco)

Código	Descrição
5	Pisca-pisca
7	Vídeo reverso (invertido)
40	Fundo preto (ou cinza)
41	Fundo vermelho
42	Fundo verde
43	Fundo marrom (ou amarelo)
44	Fundo azul
45	Fundo roxo
46	Fundo ciano
47	Fundo cinza (ou branco)



SHELL SCRIPT

na prática

Prática 01 - Testando arquivo

- Criar o script testa-arquivo.sh, que **pede ao usuário para digitar um arquivo e testa se ele existe**
- **Se o arquivo existe** deve retornar mensagem com **letras de cor verde**
- **Se o arquivo não existir** deve retornar mensagem com **letras de cor vermelha**

SHELL SCRIPT

na prática

Parâmetros

- Parâmetros enviados na linha de comando
- Exemplo:

\$./arquivo.sh param1 param2

Exemplo 05 - Mostrando valores das variáveis especiais

```
#!/bin/bash
```

```
echo "O nome deste script é: $0"
```

```
echo
```

```
echo "Recebidos $# argumentos: $*"
```

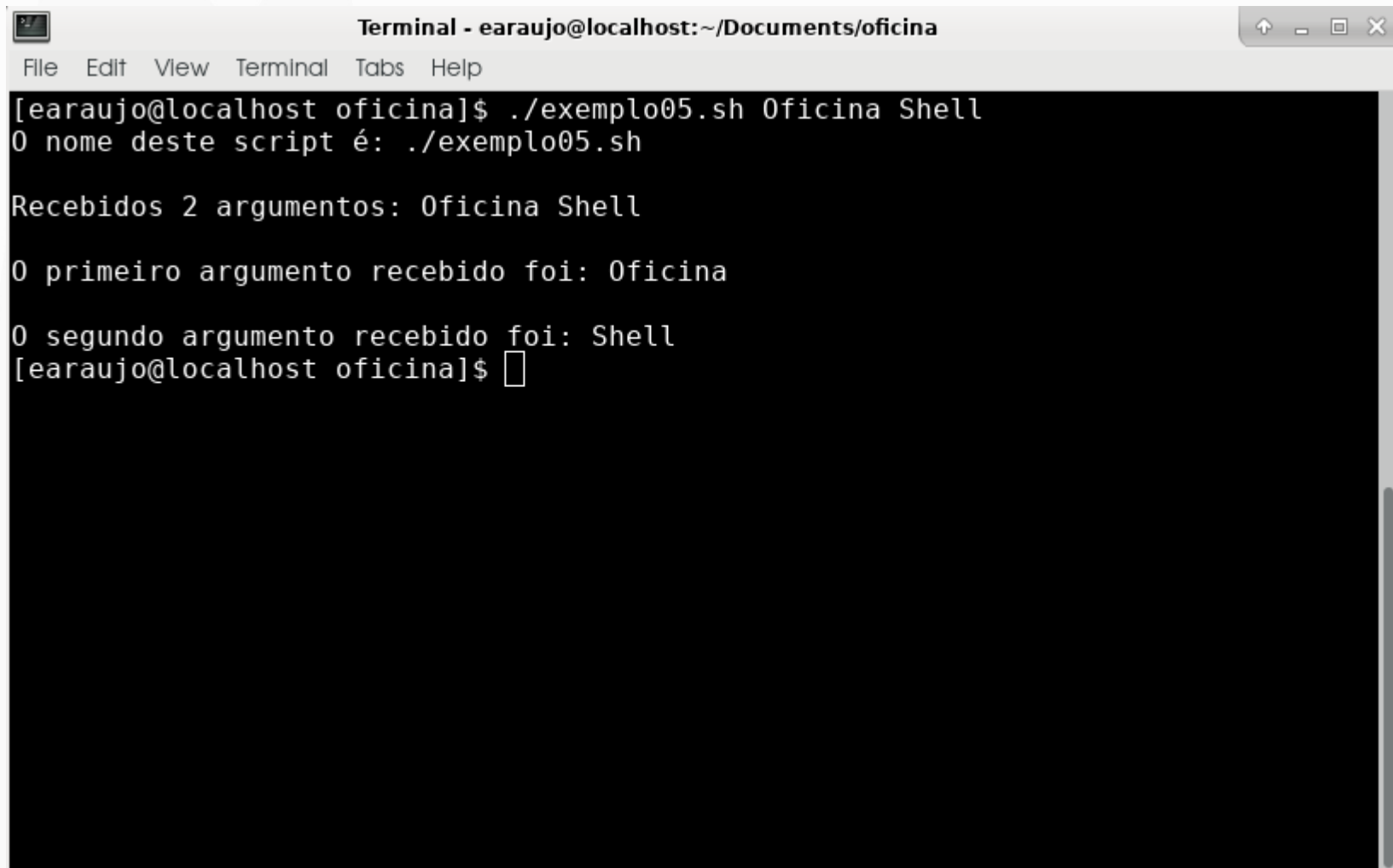
```
echo
```

```
echo "O primeiro argumento recebido foi: $1"
```

```
echo
```

```
echo "O segundo argumento recebido foi: $2"
```

Exemplo 05 - Mostrando valores das variáveis especiais

A screenshot of a terminal window titled "Terminal - earaujo@localhost:~/Documents/oficina". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows a command being executed: `[earaujo@localhost oficina]$./exemplo05.sh Oficina Shell`. The output of the script is: `0 nome deste script é: ./exemplo05.sh`, `Recebidos 2 argumentos: Oficina Shell`, `0 primeiro argumento recebido foi: Oficina`, and `0 segundo argumento recebido foi: Shell`. The prompt `[earaujo@localhost oficina]$` is followed by a cursor.

```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help
[earaujo@localhost oficina]$ ./exemplo05.sh Oficina Shell
0 nome deste script é: ./exemplo05.sh

Recebidos 2 argumentos: Oficina Shell

0 primeiro argumento recebido foi: Oficina

0 segundo argumento recebido foi: Shell
[earaujo@localhost oficina]$
```

Exemplo 06 - Mostrando os parâmetros recebidos

```
#!/bin/bash
```

```
i=0
```

```
while test "$1"
```

```
do
```

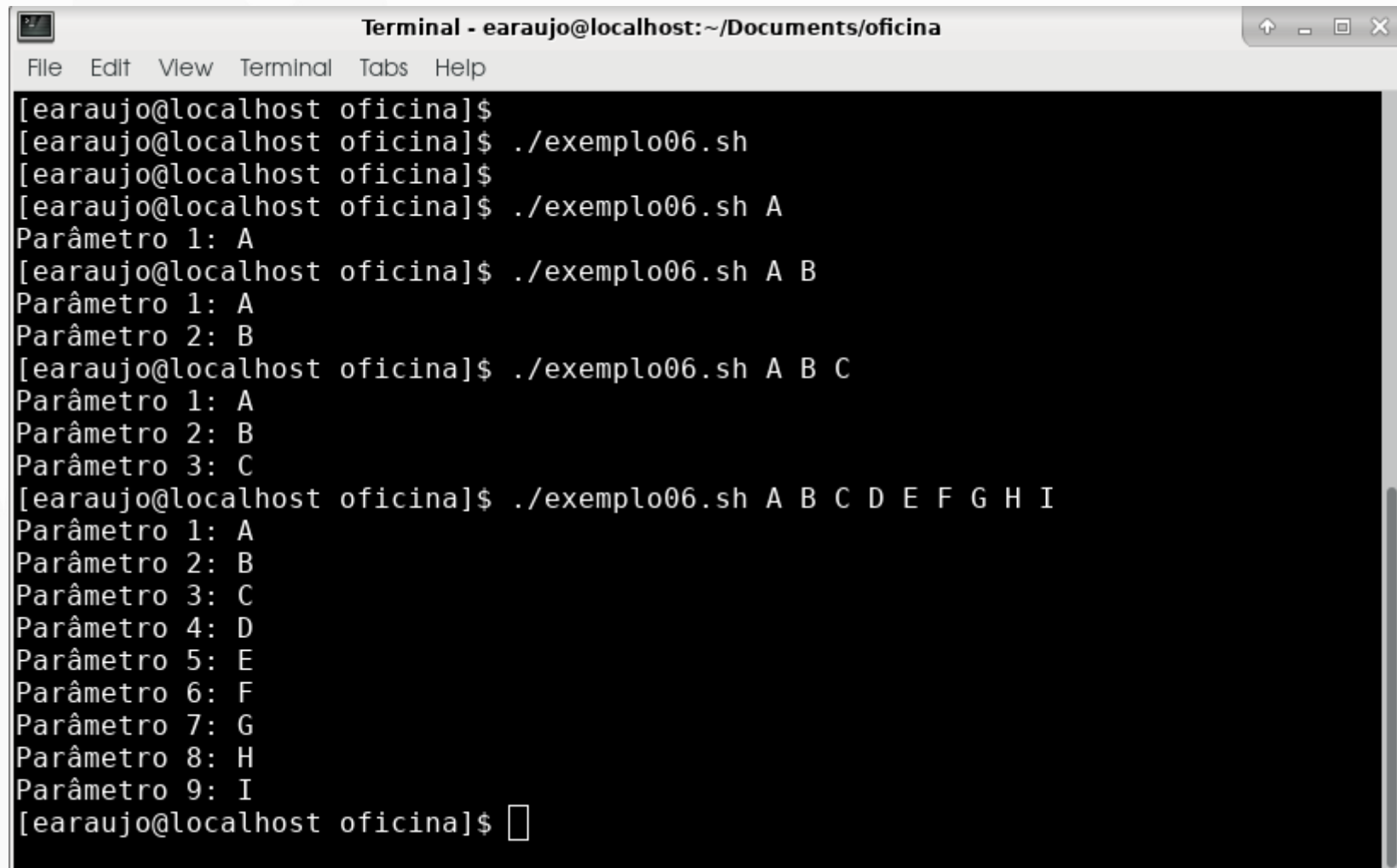
```
    i=$((i+1))
```

```
    echo "Parâmetro $i: $1 "
```

```
    shift
```

```
done
```

Exemplo 06 - Mostrando os parâmetros recebidos

A screenshot of a terminal window titled "Terminal - earaujo@localhost:~/Documents/oficina". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal shows the execution of a script named "exemplo06.sh" with varying numbers of command-line arguments. Each time the script is run, it prints the received parameters in a numbered format. The sequence of commands and outputs is as follows:

```
[earaujo@localhost oficina]$  
[earaujo@localhost oficina]$ ./exemplo06.sh  
[earaujo@localhost oficina]$  
[earaujo@localhost oficina]$ ./exemplo06.sh A  
Parâmetro 1: A  
[earaujo@localhost oficina]$ ./exemplo06.sh A B  
Parâmetro 1: A  
Parâmetro 2: B  
[earaujo@localhost oficina]$ ./exemplo06.sh A B C  
Parâmetro 1: A  
Parâmetro 2: B  
Parâmetro 3: C  
[earaujo@localhost oficina]$ ./exemplo06.sh A B C D E F G H I  
Parâmetro 1: A  
Parâmetro 2: B  
Parâmetro 3: C  
Parâmetro 4: D  
Parâmetro 5: E  
Parâmetro 6: F  
Parâmetro 7: G  
Parâmetro 8: H  
Parâmetro 9: I  
[earaujo@localhost oficina]$
```

Opções de Linha de Comando

- Opções clássicas que são utilizadas pela maioria dos programas são: -h ou --help e -v ou --version
- Exemplo:
\$ ifconfig --help
\$ df --version
\$ df --help

Exemplo 07 - Mostrando o help do script

```
#!/bin/bash
```

```
HELP=""
```

```
Uso: $0 [-h]
```

```
-h          Mostra informação de ajuda
```

```
"
```

```
if test "$1" = "-h"
```

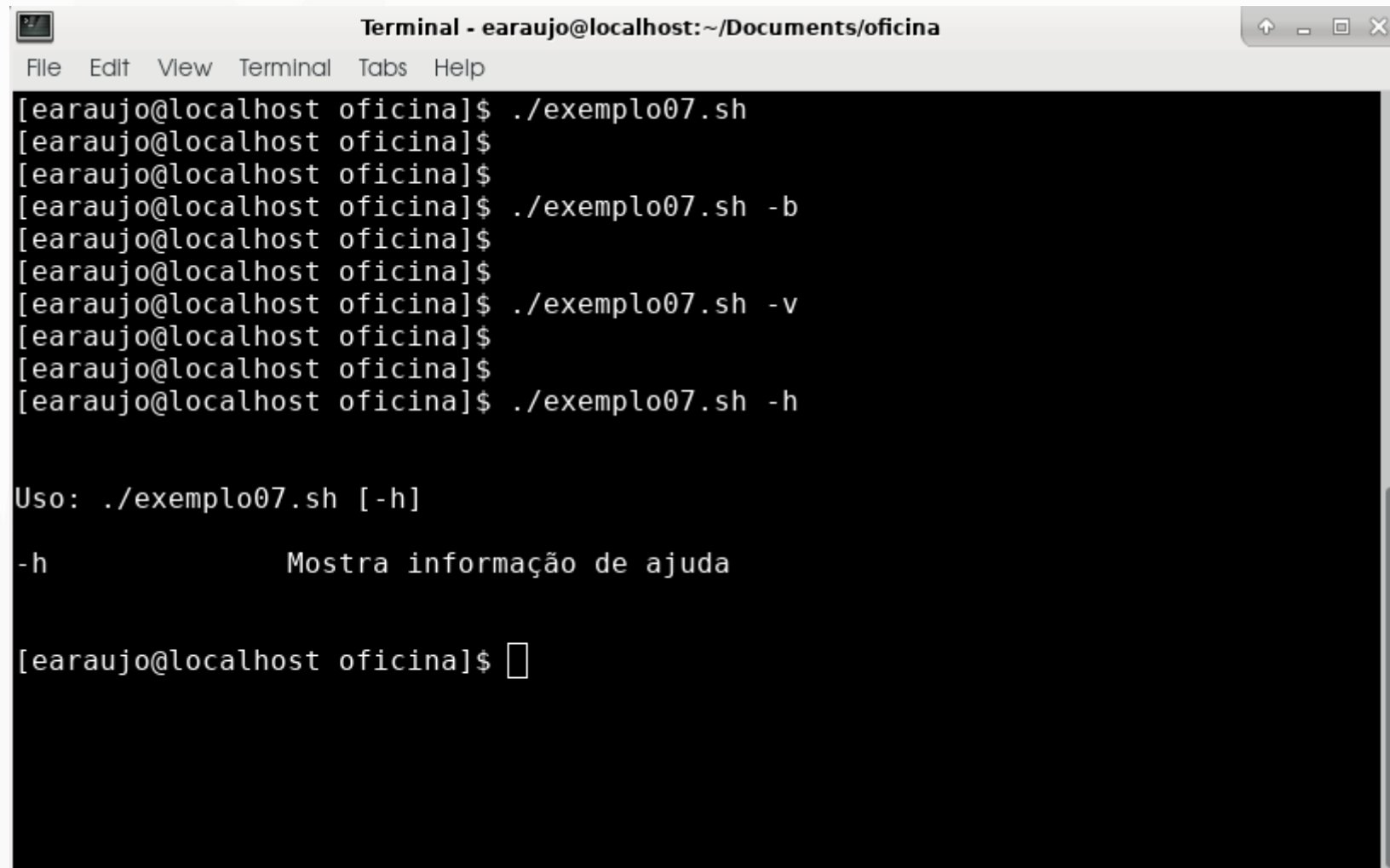
```
then
```

```
    echo "$HELP"
```

```
    exit 0
```

```
fi
```

Exemplo 07 - Mostrando o help do script



```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help
[earaujo@localhost oficina]$ ./exemplo07.sh
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo07.sh -b
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo07.sh -v
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo07.sh -h

Uso: ./exemplo07.sh [-h]

-h                Mostra informação de ajuda

[earaujo@localhost oficina]$
```

Exemplo 08 - Mostrando a versão do script

```
#!/bin/bash
```

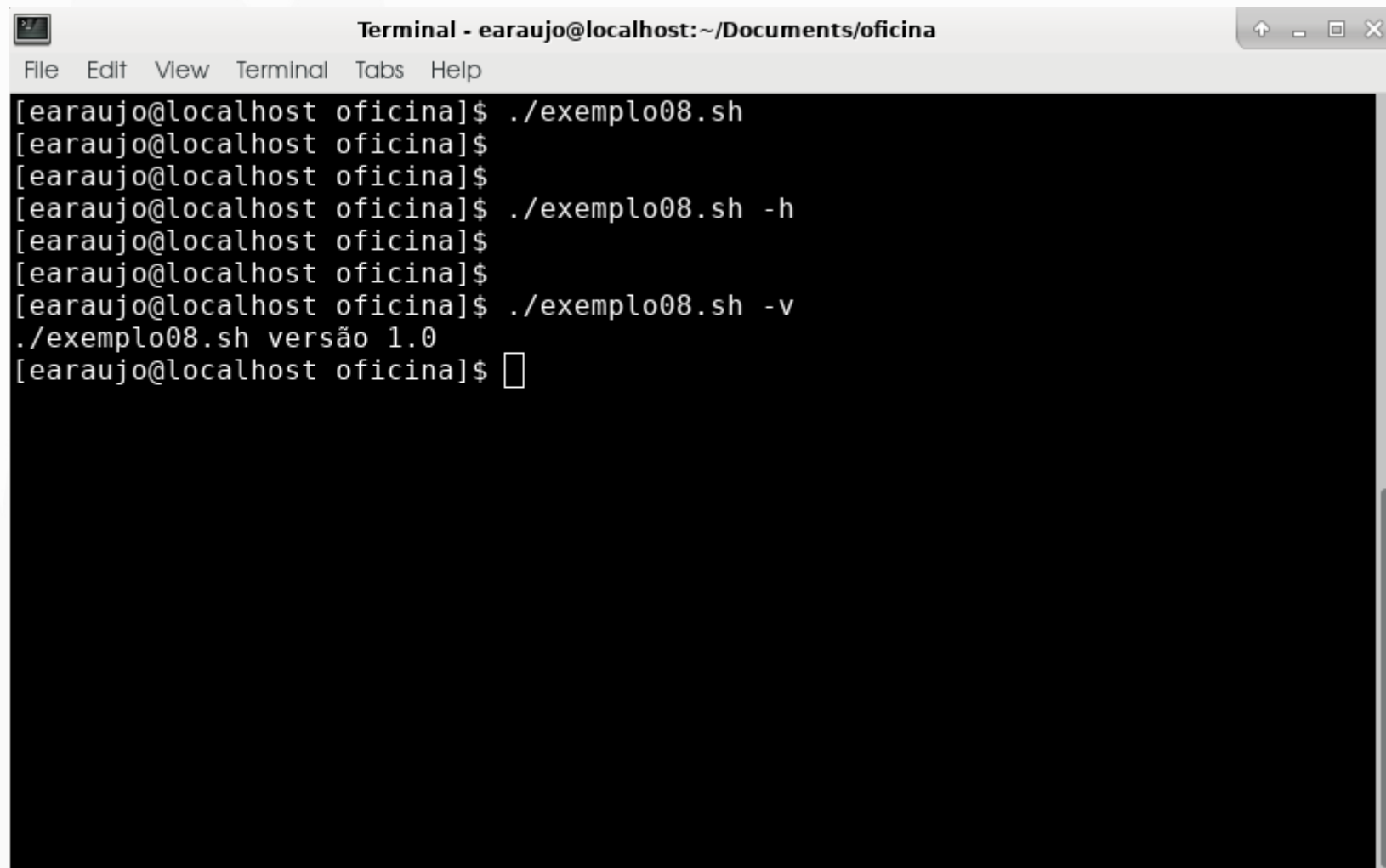
```
if test "$1" = "-v"
```

```
then
```

```
    echo "Versão do script: 1.0"
```

```
fi
```


Exemplo 08 - Mostrando a versão do script

A screenshot of a terminal window titled "Terminal - earaujo@localhost:~/Documents/oficina". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows a series of commands and their outputs. The first three lines are empty prompts. The fourth line shows the command `./exemplo08.sh -h`. The fifth line is an empty prompt. The sixth line shows the command `./exemplo08.sh -v`. The seventh line shows the output `./exemplo08.sh versão 1.0`. The eighth line is an empty prompt with a cursor. The terminal background is black with white text.

```
Terminal - earaujo@localhost:~/Documents/oficina
File Edit View Terminal Tabs Help
[earaujo@localhost oficina]$ ./exemplo08.sh
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo08.sh -h
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$
[earaujo@localhost oficina]$ ./exemplo08.sh -v
./exemplo08.sh versão 1.0
[earaujo@localhost oficina]$
```



SHELL SCRIPT

na prática

Prática 02 - Instalando programa

- Criar script instala-programa.sh que **instala um programa recebido como parâmetro**
- Retornar **mensagem de versão ou ajuda** utilizando as **opções -v e -h** por linha de comando

SHELL SCRIPT

na prática

Dialog

- Programa de console em modo texto que **desenha janelas**
- **Interfaces gráficas amigáveis** para o usuário
- Feito para ser utilizado com shell

Exemplo 09 - Mostrando mensagem

```
#!/bin/bash
```

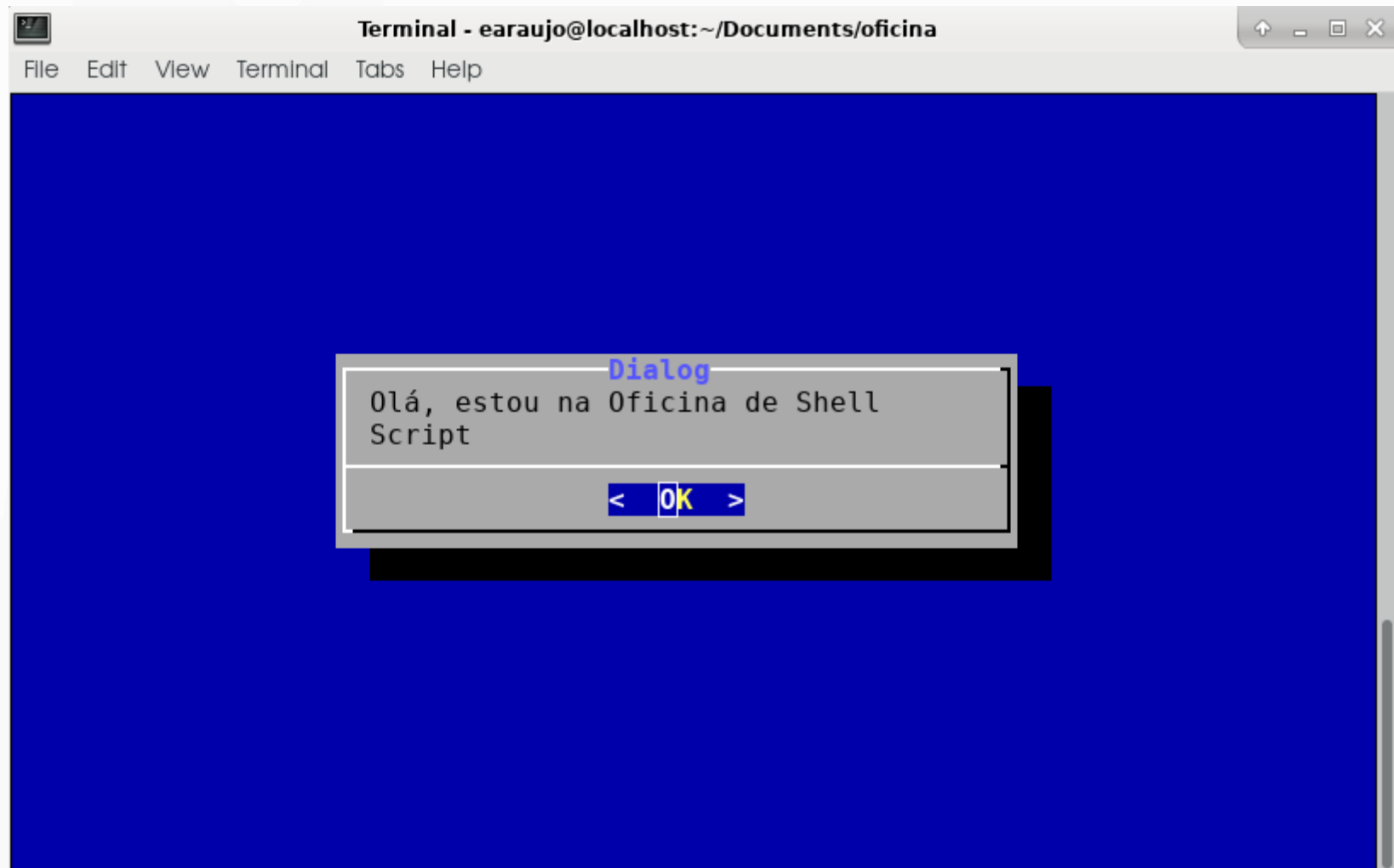
```
dialog
```

```
--title "Dialog"
```

```
--msgbox "Olá, estou na Oficina de Shell Script"
```

```
6 40
```

Exemplo 09 - Mostrando mensagem



Exemplo 010 - Monitorando uso de disco

```
#!/bin/bash
```

```
df > out &
```

```
dialog \
```

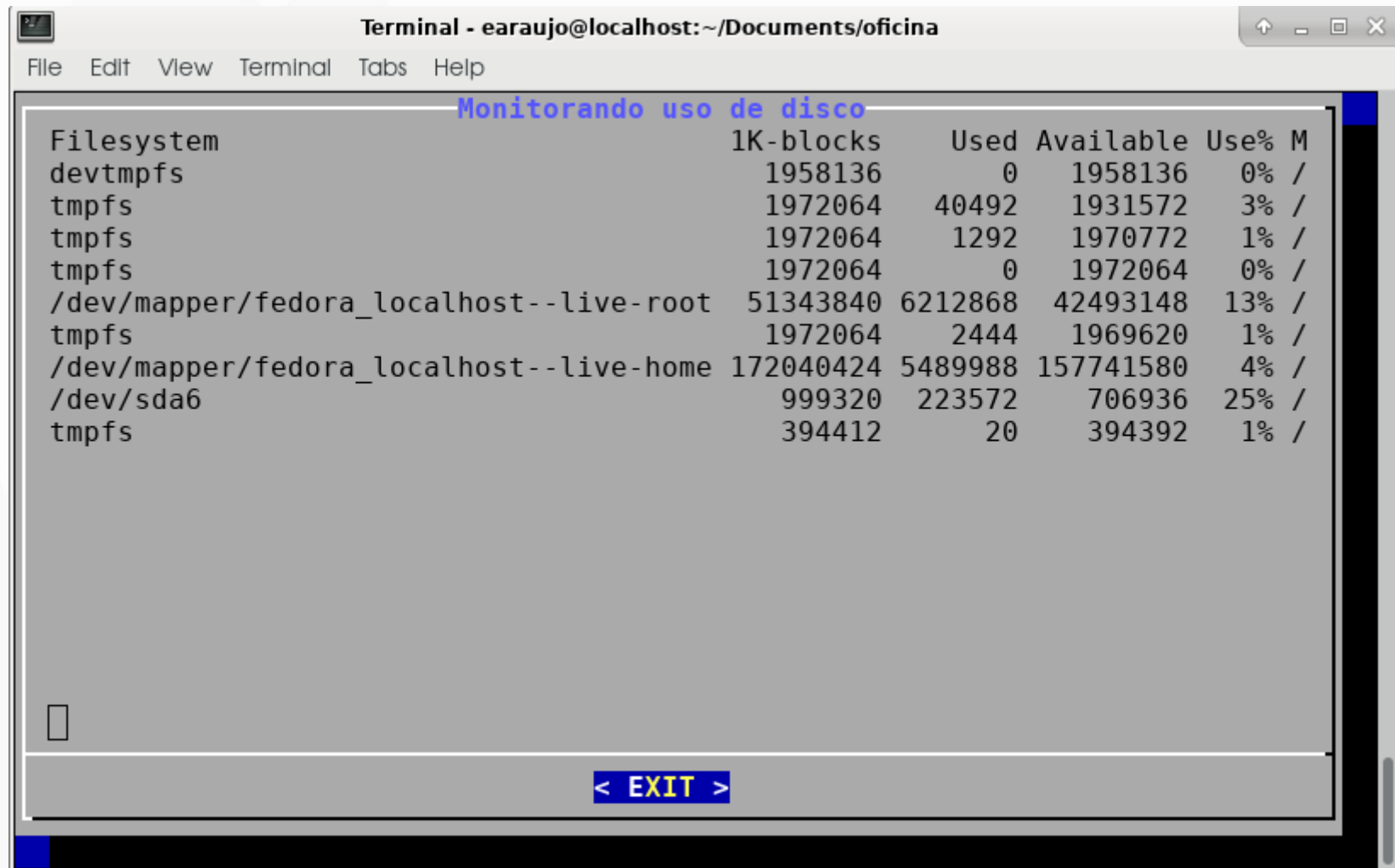
```
--title "Monitorando uso de disco" \
```

```
--tailbox out \
```

```
0 0
```

```
clear
```


Exemplo 010 - Monitorando uso de disco



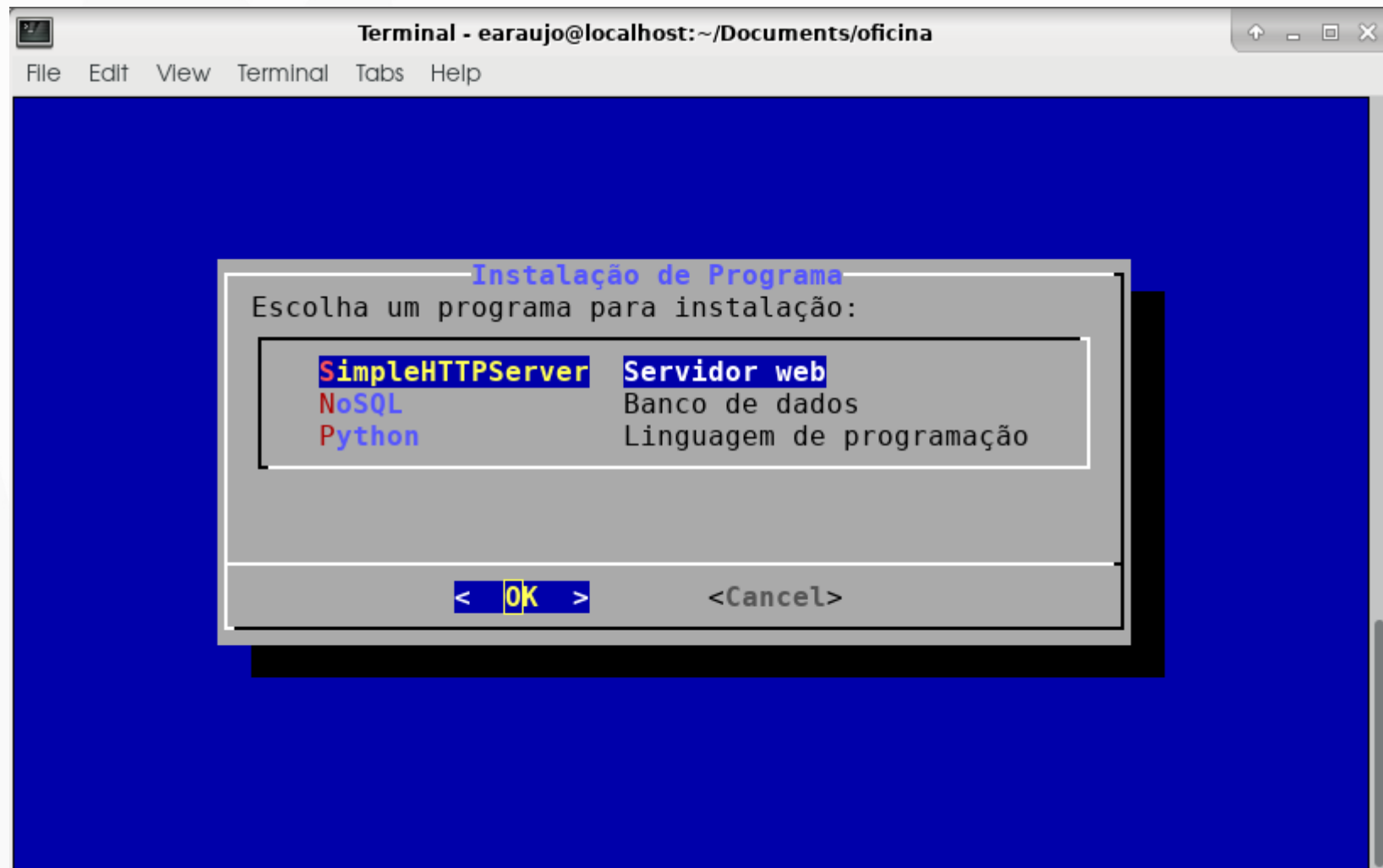
A terminal window titled "Terminal - earaujo@localhost:~/Documents/oficina" displays the output of the `df` command. The output is a table showing disk usage for various filesystems. The title "Monitorando uso de disco" is displayed in blue. At the bottom of the terminal, there is a blue button with the text "< EXIT >".

Filesystem	1K-blocks	Used	Available	Use%	M
devtmpfs	1958136	0	1958136	0%	/
tmpfs	1972064	40492	1931572	3%	/
tmpfs	1972064	1292	1970772	1%	/
tmpfs	1972064	0	1972064	0%	/
/dev/mapper/fedora_localhost--live-root	51343840	6212868	42493148	13%	/
tmpfs	1972064	2444	1969620	1%	/
/dev/mapper/fedora_localhost--live-home	172040424	5489988	157741580	4%	/
/dev/sda6	999320	223572	706936	25%	/
tmpfs	394412	20	394392	1%	/

Exemplo 011 - Mostrando um menu

```
#!/bin/bash
dialog \
  --title "Instalação de Programa" \
  --menu "Escolha um programa para instalação: " \
  0 0 0 \
  SimpleHTTPServer "Servidor web" \
  NoSQL "Banco de dados" \
  Python "Linguagem de programação"
clear
```

Exemplo 011 - Mostrando menu



Script de instalação de programas com o Dialog

Resumo

- Arquivo Shell Script
- Variáveis
- If, for e while
- Interação com usuário
- Comando test
- Cores
- Parâmetros
- Opções por linha de comando
- Dialog

SHELL SCRIPT

na prática

EDSON ARAÚJO

Email: edsonlead@gmail.com

Blog: edsonlead.com

LinkedIn: [Edson Araújo](#)

Twitter: [@earaujopy](#)