

Introdução à Ciência da Computação

Shell Script – parte II

Professor Iago Augusto de Carvalho
iago.carvalho@unifal-mg.edu.br

Criando scripts simples

Múltiplos comandos

É possível encadear múltiplos comandos em sequência no bash, bastando para isso separá-los com um ponto-e-vírgula:

`cd /; ls -l`

Essa sequência pode ser considerada como um script do shell ultra simples.

Claro que toda vez que for necessária a execução dessa sequência, deve-se digitar todos os comandos novamente.

Para evitar esse problema, criaremos arquivos de script contendo os comandos que queremos executar.

Criando scripts simples

Arquivo de Script

Para criar um arquivo de script, use um editor de texto de sua preferência (vi, nano, emacs, gedit) e crie um arquivo.

A primeira tarefa é especificar qual shell será usado na primeira linha do arquivo:

`#!/bin/bash`

O símbolo **`#`** representa uma linha de comentário, exceto na primeira linha do arquivo, onde a combinação **`#!`** indica o shell que rodará o script.

Criando scripts simples

Arquivo de Script

Após especificar o shell podemos entrar com os comandos a serem executados, juntamente com os comentários necessários:

```
#!/bin/bash  
# Primeiro script no bash  
cd /  
ls -l
```

O shell processará os comandos na ordem em que aparecem no arquivo.

Criando scripts simples

Arquivo de Script

```
adriana@adriana-VirtualBox: ~  
1 adriana@adriana-VirtualBox:~$ gedit meuscript.sh  
3 adriana@adriana-VirtualBox:~$ ls me*  
meuscript.sh  
adriana@adriana-VirtualBox:~$
```

```
Abrir ▼ meuscript.sh Salvar  
2 1 #!/bin/bash  
2 # Primeiro script  
3 cd /  
4 ls -l
```

Ainda não executa.

Criando scripts simples

Arquivo de Script

Após digitar os comandos desejados no arquivo de script, salve-o com o nome desejado.

É necessário dar permissão de execução ao script, caso contrário ele não roda:

chmod a+x meuscript.sh


Opcionalmente, adicione também o script a algum diretório presente na variável PATH, ou adicione o diretório que contém a variável.

Rode o script chamando-o pelo nome:

meuscript.sh

Criando scripts simples


Arquivo de Script



```
adriana@adriana-VirtualBox:~$ ls -l
total 36
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 'Área de Trabalho'
drwxr-xr-x 2 adriana adriana 4096 jul  9 11:26 Documentos
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Downloads
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Imagens
-rw-rw-r-- 1 adriana adriana  41 jul 14 14:26 meuscript.sh
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Modelos
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Música
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Público
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Vídeos
adriana@adriana-VirtualBox:~$
```

Criando scripts simples

Arquivo de Script



```
adriana@adriana-VirtualBox:~$ chmod a+x meuscript.sh
adriana@adriana-VirtualBox:~$ ls -l
total 36
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 'Área de Trabalho'
drwxr-xr-x 2 adriana adriana 4096 jul  9 11:26 Documentos
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Downloads
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Imagens
-rwxrwxr-x 1 adriana adriana  41 jul 14 14:26 meuscript.sh
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Modelos
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Música
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Público
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Vídeos
adriana@adriana-VirtualBox:~$
```

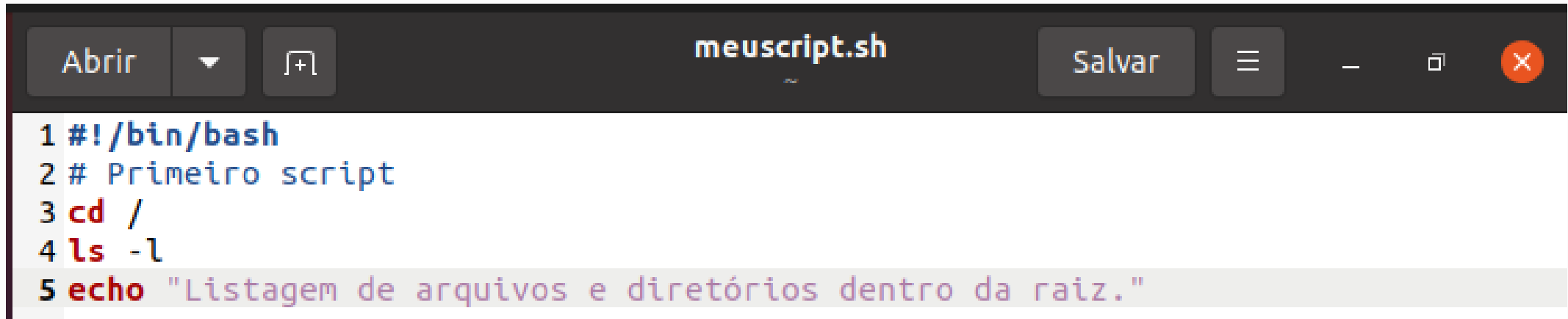

Criando scripts simples

Arquivo de Script

```
adriana@adriana-VirtualBox:~$ ./meuscript.sh
total 76
lrwxrwxrwx    1 root root      7 jun 11 12:36 bin -> usr/bin
drwxr-xr-x    3 root root  4096 jun 17 12:09 boot
drwxrwxr-x    2 root root  4096 jun 11 12:38 cdrom
drwxr-xr-x   19 root root  4100 jul 14 12:05 dev
drwxr-xr-x  130 root root 12288 jul  9 14:09 etc
drwxr-xr-x    3 root root  4096 jun 11 12:40 home
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib -> usr/lib
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib32 -> usr/lib32
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib64 -> usr/lib64
lrwxrwxrwx    1 root root     10 jun 11 12:36 libx32 -> usr/libx32
drwx-----   2 root root 16384 jun 11 12:36 lost+found
drwxr-xr-x    2 root root  4096 fev  9 15:47 media
drwxr-xr-x    2 root root  4096 fev  9 15:47 mnt
drwxr-xr-x    2 root root  4096 fev  9 15:47 opt
dr-xr-xr-x  234 root root      0 jul 14 12:04 proc
drwx-----   4 root root  4096 jun 11 12:49 root
drwxr-xr-x   29 root root   780 jul 14 12:05 run
lrwxrwxrwx    1 root root      8 jun 11 12:36 sbin -> usr/sbin
drwxr-xr-x    8 root root  4096 fev  9 15:57 snap
drwxr-xr-x    2 root root  4096 fev  9 15:47 srv
dr-xr-xr-x   13 root root      0 jul 14 12:04 sys
drwxrwxrwt   18 root root  4096 jul 14 14:56 tmp
drwxr-xr-x   14 root root  4096 fev  9 15:48 usr
drwxr-xr-x   14 root root  4096 fev  9 15:56 var
adriana@adriana-VirtualBox:~$
```

Criando scripts simples

Arquivo de Script



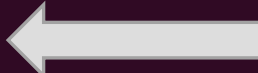
The image shows a terminal window with a dark header bar. The header contains the title 'meuscript.sh' in the center, and on the left, buttons for 'Abrir' (Open), a dropdown arrow, and a '+' icon. On the right, there are buttons for 'Salvar' (Save), a hamburger menu icon, a '-' icon, a window icon, and a red 'X' close button. The main area of the terminal is white and contains a script with five lines, each numbered. The script is as follows:

```
1 #!/bin/bash
2 # Primeiro script
3 cd /
4 ls -l
5 echo "Listagem de arquivos e diretórios dentro da raiz."
```

Criando scripts simples

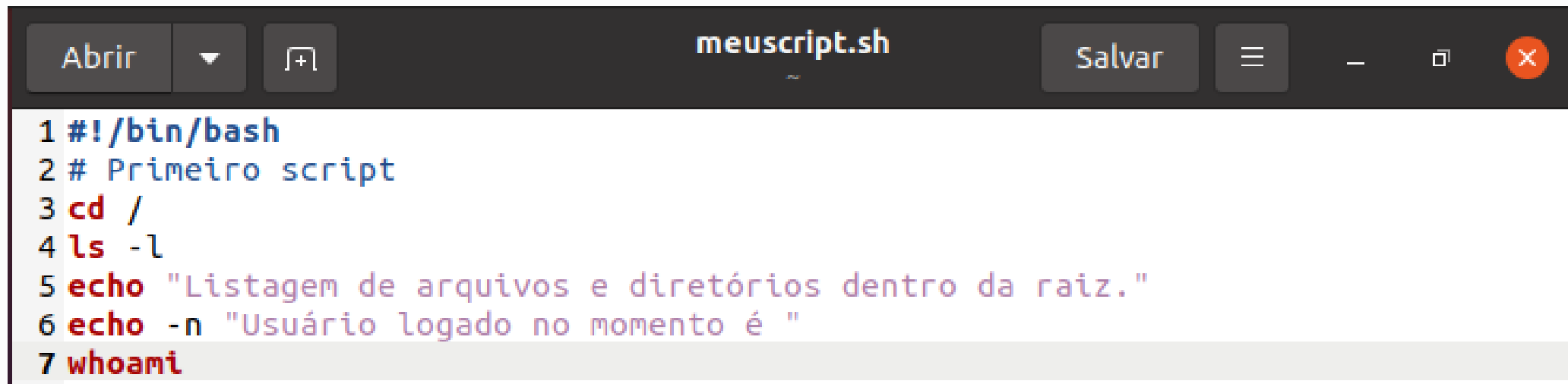
Arquivo de Script

```
adriana@adriana-VirtualBox:~$ gedit meuscript.sh
adriana@adriana-VirtualBox:~$ ./meuscript.sh
total 76
lrwxrwxrwx    1 root root      7 jun 11 12:36 bin -> usr/bin
drwxr-xr-x    3 root root  4096 jun 17 12:09 boot
drwxrwxr-x    2 root root  4096 jun 11 12:38 cdrom
drwxr-xr-x   19 root root  4100 jul 14 12:05 dev
drwxr-xr-x  130 root root 12288 jul  9 14:09 etc
drwxr-xr-x    3 root root  4096 jun 11 12:40 home
lrwxrwxrwx    1 root root      7 jun 11 12:36 lib -> usr/lib
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib32 -> usr/lib32
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib64 -> usr/lib64
lrwxrwxrwx    1 root root     10 jun 11 12:36 libx32 -> usr/libx32
drwx-----   2 root root 16384 jun 11 12:36 lost+found
drwxr-xr-x    2 root root  4096 fev  9 15:47 media
drwxr-xr-x    2 root root  4096 fev  9 15:47 mnt
drwxr-xr-x    2 root root  4096 fev  9 15:47 opt
dr-xr-xr-x  232 root root      0 jul 14 12:04 proc
drwx-----   4 root root  4096 jun 11 12:49 root
drwxr-xr-x   29 root root   780 jul 14 12:05 run
lrwxrwxrwx    1 root root      8 jun 11 12:36 sbin -> usr/sbin
drwxr-xr-x    8 root root  4096 fev  9 15:57 snap
drwxr-xr-x    2 root root  4096 fev  9 15:47 srv
dr-xr-xr-x   13 root root      0 jul 14 12:04 sys
drwxrwxrwt   17 root root  4096 jul 14 14:56 tmp
drwxr-xr-x   14 root root  4096 fev  9 15:48 usr
drwxr-xr-x   14 root root  4096 fev  9 15:56 var
Listagem de arquivos e diretórios dentro da raiz.
adriana@adriana-VirtualBox:~$
```



Criando scripts simples

Arquivo de Script



The image shows a code editor window with a dark theme. The title bar at the top reads 'meuscript.sh'. On the left side of the title bar are buttons for 'Abrir' (Open) and a dropdown menu. On the right side are buttons for 'Salvar' (Save), a hamburger menu, a window management icon, and a red close button. The main area of the editor contains a shell script with seven lines of code, each preceded by a line number. The code is as follows:

```
1 #!/bin/bash
2 # Primeiro script
3 cd /
4 ls -l
5 echo "Listagem de arquivos e diretórios dentro da raiz."
6 echo -n "Usuário logado no momento é "
7 whoami
```

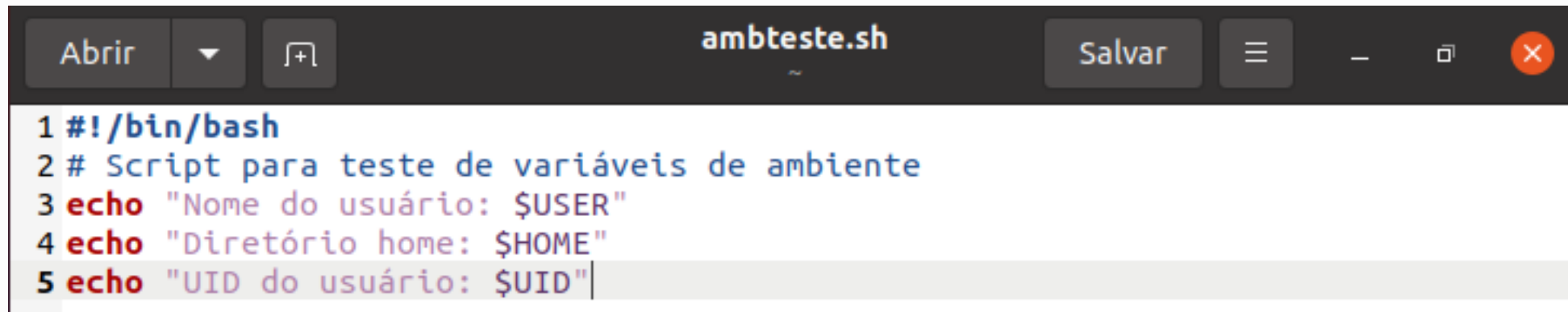
Criando scripts simples

Arquivo de Script

```
adriana@adriana-VirtualBox:~$ ./meuscript.sh
total 76
lrwxrwxrwx    1 root root      7 jun 11 12:36 bin -> usr/bin
drwxr-xr-x    3 root root  4096 jun 17 12:09 boot
drwxrwxr-x    2 root root  4096 jun 11 12:38 cdrom
drwxr-xr-x   19 root root  4100 jul 14 12:05 dev
drwxr-xr-x  130 root root 12288 jul  9 14:09 etc
drwxr-xr-x    3 root root  4096 jun 11 12:40 home
lrwxrwxrwx    1 root root      7 jun 11 12:36 lib -> usr/lib
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib32 -> usr/lib32
lrwxrwxrwx    1 root root      9 jun 11 12:36 lib64 -> usr/lib64
lrwxrwxrwx    1 root root     10 jun 11 12:36 libx32 -> usr/libx32
drwx-----   2 root root 16384 jun 11 12:36 lost+found
drwxr-xr-x    2 root root  4096 fev  9 15:47 media
drwxr-xr-x    2 root root  4096 fev  9 15:47 mnt
drwxr-xr-x    2 root root  4096 fev  9 15:47 opt
dr-xr-xr-x  231 root root      0 jul 14 12:04 proc
drwx-----   4 root root  4096 jun 11 12:49 root
drwxr-xr-x   29 root root   780 jul 14 12:05 run
lrwxrwxrwx    1 root root      8 jun 11 12:36 sbin -> usr/sbin
drwxr-xr-x    8 root root  4096 fev  9 15:57 snap
drwxr-xr-x    2 root root  4096 fev  9 15:47 srv
dr-xr-xr-x   13 root root      0 jul 14 12:04 sys
drwxrwxrwt   17 root root  4096 jul 14 14:56 tmp
drwxr-xr-x   14 root root  4096 fev  9 15:48 usr
drwxr-xr-x   14 root root  4096 fev  9 15:56 var
Listagem de arquivos e diretórios dentro da raiz.
Usuário logado no momento é adriana
adriana@adriana-VirtualBox:~$
```

Usando variáveis de Ambiente

```
#!/bin/bash
# Informações sobre o usuário:
echo "Usuário: $USER"
echo "Diretório home: $HOME"
echo "UID do usuário: "$UID"
```



The screenshot shows a code editor window with a dark theme. The title bar at the top contains the filename 'ambteste.sh' and several icons: a folder icon, a dropdown arrow, a plus icon, a 'Salvar' (Save) button, a hamburger menu icon, a minus icon, a square icon, and a red close button. The editor area displays a shell script with five lines of code, each preceded by a line number. The code is syntax-highlighted: the shebang is blue, comments are blue, and the 'echo' command is red. The script prints the values of the environment variables \$USER, \$HOME, and \$UID.

```
1 #!/bin/bash
2 # Script para teste de variáveis de ambiente
3 echo "Nome do usuário: $USER"
4 echo "Diretório home: $HOME"
5 echo "UID do usuário: $UID"
```

Usando variáveis de Ambiente

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ gedit ambteste.sh  
adriana@adriana-VirtualBox:~$ ./ambteste.sh  
bash: ./ambteste.sh: Permissão negada  
adriana@adriana-VirtualBox:~$
```

```
adriana@adriana-VirtualBox:~$ chmod a+x ambteste.sh  
adriana@adriana-VirtualBox:~$ ./ambteste.sh  
Nome do usuário: adriana  
Diretório home: /home/adriana  
UID do usuário: 1000  
adriana@adriana-VirtualBox:~$
```

Usando variáveis de Usuário

Variáveis próprias podem ser criadas dentro de um script.

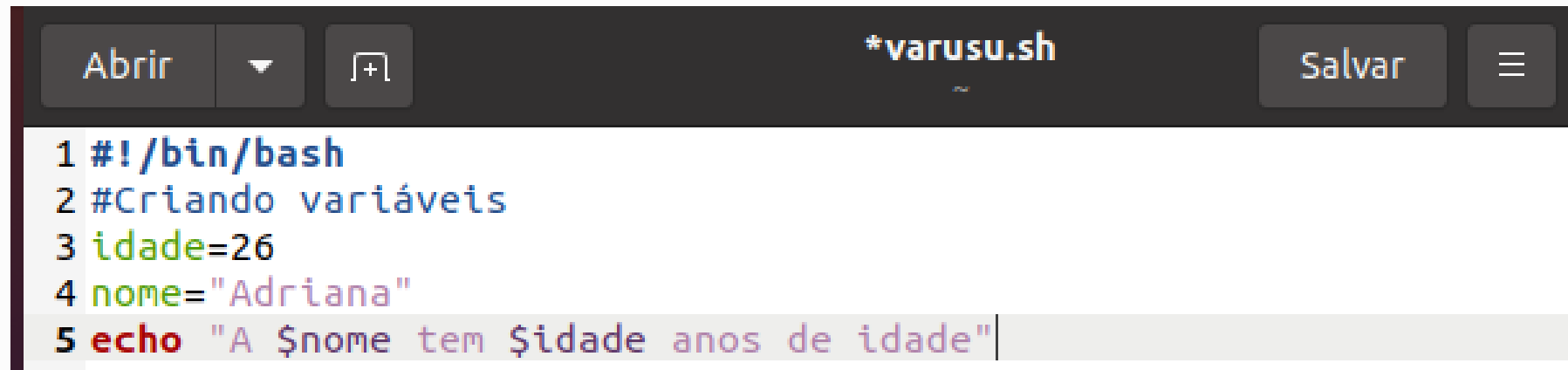
Variáveis de usuário podem conter até 20 caracteres no nome e são case-sensitive.

O shell determina automaticamente o tipo das variáveis criadas.

As variáveis criadas pelo usuário são desalocadas quando termina a execução do script.

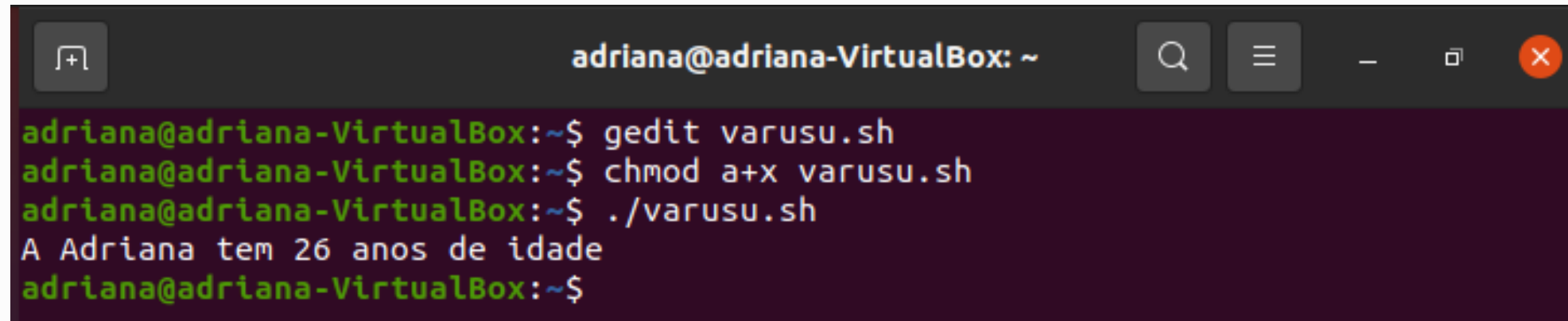
Criando variáveis de Usuário

```
#!/bin/bash  
# Criando variáveis  
idade=26  
nome="Adriana"  
echo "A $nome tem $idade anos de idade"
```



```
*varusu.sh  
1 #!/bin/bash  
2 #Criando variáveis  
3 idade=26  
4 nome="Adriana"  
5 echo "A $nome tem $idade anos de idade"
```

Criando variáveis de Usuário

A terminal window titled 'adriana@adriana-VirtualBox: ~' with standard window controls. The terminal shows a sequence of commands and their output. The user runs 'gedit varusu.sh' to create a file, 'chmod a+x varusu.sh' to make it executable, and './varusu.sh' to run it. The script outputs 'A Adriana tem 26 anos de idade'.

```
adriana@adriana-VirtualBox:~$ gedit varusu.sh
adriana@adriana-VirtualBox:~$ chmod a+x varusu.sh
adriana@adriana-VirtualBox:~$ ./varusu.sh
A Adriana tem 26 anos de idade
adriana@adriana-VirtualBox:~$
```

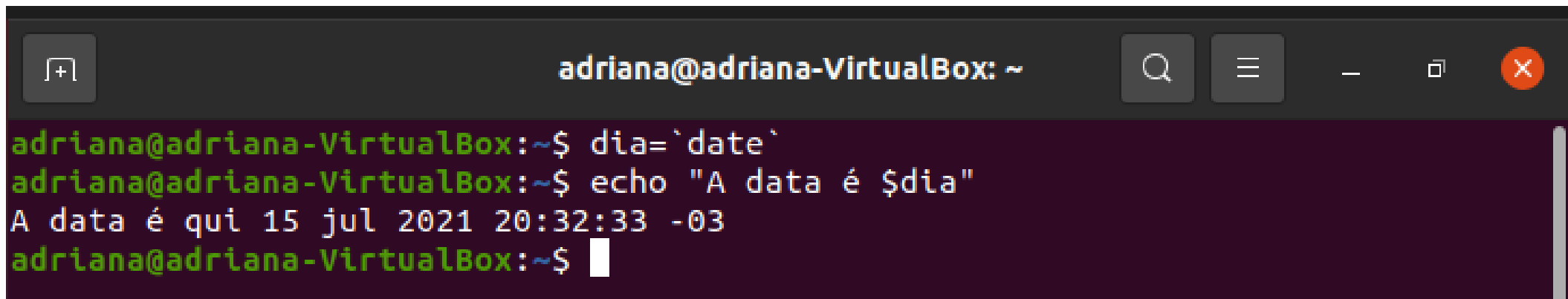
Redirecionar saída de comandos e variáveis

Usando crases (backtick)

As crases permitem atribuir a saída de um comando do shell a uma variável.

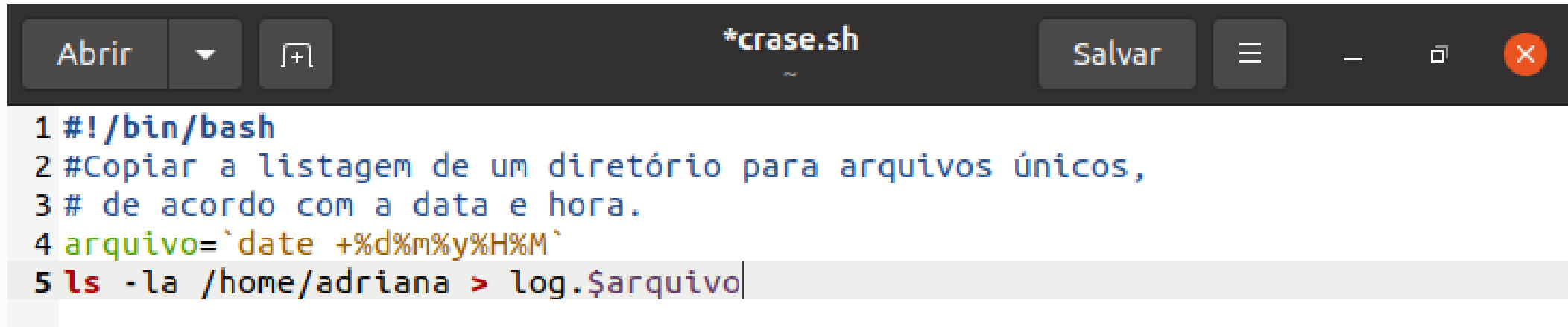
O comando todo deve estar entre crases:

```
dia=`date`  
echo "A data é $dia"
```



```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ dia=`date`  
adriana@adriana-VirtualBox:~$ echo "A data é $dia"  
A data é qui 15 jul 2021 20:32:33 -03  
adriana@adriana-VirtualBox:~$
```

Redirecionar saída de comandos e variáveis



```
Abrir  *craze.sh  Salvar  [Menu Icon]  [Window Icon]  [Close Icon]

1 #!/bin/bash
2 #Copiar a listagem de um diretório para arquivos únicos,
3 # de acordo com a data e hora.
4 arquivo=`date +%d%m%y%H%M`
5 ls -la /home/adriana > log.$arquivo
```

Redirecionar saída de comandos e variáveis

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ gedit crase.sh  
adriana@adriana-VirtualBox:~$ chmod 755 crase.sh  
adriana@adriana-VirtualBox:~$ ls  
  ambteste.sh      Documentos  meuscript.sh  Público  
'Área de Trabalho' Downloads  Modelos      varusu.sh  
  crase.sh         Imagens    Música        Vídeos  
adriana@adriana-VirtualBox:~$ ./crase.sh  
adriana@adriana-VirtualBox:~$ ls  
  ambteste.sh      Documentos  log.1507212038  Música      Vídeos  
'Área de Trabalho' Downloads  meuscript.sh    Público  
  crase.sh         Imagens    Modelos        varusu.sh  
adriana@adriana-VirtualBox:~$
```

Redirecionar saída de comandos e variáveis

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ gedit crase.sh  
adriana@adriana-VirtualBox:~$ chmod 755 crase.sh  
adriana@adriana-VirtualBox:~$ ls  
ambteste.sh      Documentos      meuscript.sh    Público  
'Área de Trabalho' Downloads      Modelos         varusu.sh  
crase.sh         Imagens        Música          Vídeos  
adriana@adriana-VirtualBox:~$ ./crase.sh  
adriana@adriana-VirtualBox:~$ ls  
ambteste.sh      Documentos      log.1507212038  Música        Vídeos  
'Área de Trabalho' Downloads      meuscript.sh    Público  
crase.sh         Imagens        Modelos         varusu.sh  
adriana@adriana-VirtualBox:~$ ./crase.sh  
adriana@adriana-VirtualBox:~$ ls  
ambteste.sh      Documentos      log.1507212038  Modelos       varusu.sh  
'Área de Trabalho' Downloads      log.1507212039  Música        Vídeos  
crase.sh         Imagens        meuscript.sh    Público  
adriana@adriana-VirtualBox:~$
```

Redirecionar saída de comandos e variáveis

```
adriana@adriana-VirtualBox:~$ vi log.1507212038
adriana@adriana-VirtualBox:~$
```

```
total 316
drwxr-xr-x 17 adriana adriana 4096 jul 15 20:38 .
drwxr-xr-x  3 root    root    4096 jun 11 12:40 ..
-rwxrwxr-x  1 adriana adriana  148 jul 14 15:22 ambteste.sh
drwxr-xr-x  2 adriana adriana 4096 jun 11 12:49 Área de Trabalho
-rw-----  1 adriana adriana 3621 jul 14 15:05 .bash_history
-rw-r--r--  1 adriana adriana  220 jun 11 12:40 .bash_logout
-rw-r--r--  1 adriana adriana 3794 jul  9 15:09 .bashrc
-rw-r--r--  1 adriana adriana 20480 jul  9 12:59 .bashrc.swf
-rw-r--r--  1 adriana adriana 20480 jul  9 12:59 .bashrc.swg
-rw-r--r--  1 adriana adriana 16384 jul  9 12:57 .bashrc.swh
-rw-r--r--  1 adriana adriana 16384 jul  9 12:57 .bashrc.swi
-rw-r--r--  1 adriana adriana 20480 jul  9 12:56 .bashrc.swj
-rw-r--r--  1 adriana adriana 20480 jul  9 12:54 .bashrc.swk
-rw-r--r--  1 adriana adriana 16384 jul  9 12:53 .bashrc.swl
-rw-r--r--  1 adriana adriana 16384 jul  9 12:36 .bashrc.swm
-rw-r--r--  1 adriana adriana 20480 jul  9 12:32 .bashrc.swn
-rw-r--r--  1 adriana adriana 16384 jul  9 12:29 .bashrc.swo
-rw-r--r--  1 adriana adriana 16384 jul  9 11:42 .bashrc.swp
drwx----- 16 adriana adriana 4096 jul  9 11:26 .cache
drwx----- 12 adriana adriana 4096 jun 11 13:36 .config
-rwxr-xr-x  1 adriana adriana  166 jul 15 20:38 crase.sh
drwxr-xr-x  2 adriana adriana 4096 jul  9 11:26 Documentos
drwxr-xr-x  2 adriana adriana 4096 jun 27 14:49 Downloads
drwx-----  3 adriana adriana 4096 jul 14 14:20 .gnupg
drwxr-xr-x  2 adriana adriana 4096 jun 27 14:49 Imagens
-rw-----  1 adriana adriana   28 jul  8 20:24 .lessht
drwx-----  3 adriana adriana 4096 jun 11 12:49 .local
"log.1507212038" 42 lines, 2460 characters
```

Redirecionamento de Entrada e Saída

Redirecionamento de Saída

O mais comum é enviar a saída de um comando para um arquivo. Para isso, use o símbolo: >

comando > arquivo_saída
ls > arquivo.txt

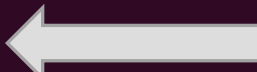
Caso queira acrescentar a saída de um comando a um arquivo em vez de sobrescrevê-lo, use o símbolo duplo: >>

comando >> arquivo_saída
date >> arquivo.txt

Redirecionamento de Entrada e Saída

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ ls > teste.txt  
adriana@adriana-VirtualBox:~$ cat teste.txt  
ambteste.sh  
Área de Trabalho  
crase.sh  
Documentos  
Downloads  
Imagens  
log.1507212038  
log.1507212039  
meuscript.sh  
Modelos  
Música  
Público  
teste.txt  
varusu.sh  
Vídeos  
adriana@adriana-VirtualBox:~$
```

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ date >> teste.txt  
adriana@adriana-VirtualBox:~$ cat teste.txt  
ambteste.sh  
Área de Trabalho  
crase.sh  
Documentos  
Downloads  
Imagens  
log.1507212038  
log.1507212039  
meuscript.sh  
Modelos  
Música  
Público  
teste.txt  
varusu.sh  
Vídeos  
qui 15 jul 2021 20:50:24 -03  
adriana@adriana-VirtualBox:~$
```



Redirecionamento de Entrada e Saída

Redirecionamento de Entrada

Em vez de redirecionar a saída de um comando para um arquivo, é possível tomar o conteúdo de um arquivo e redirecioná-lo para um comando.

Para isso, use o símbolo: <

comando < arquivo_entrada

wc < arquivo.txt

O comando **wc** produz contagem de texto do arquivo. Por padrão mostra três valores:

- Número de linhas no texto;
- Número de palavras;
- Número de bytes.

```
adriana@adriana-VirtualBox:~$ wc < teste.txt
16  23 193
adriana@adriana-VirtualBox:~$
```

Redirecionamento de Entrada e Saída

Redirecionamento de Entrada Inline

Com este método é possível especificar os dados para redirecionamento na linha de comando em vez de um arquivo.

O símbolo de redirecionamento de entrada inline é o <<

Deve-se especificar um marcador de texto que delimite o início e o fim dos dados usados para entrada.


```
wc << OLA  
> primeira frase  
> segunda frase  
> OLA
```

Marcador (pode
ser qualquer
palavra)

Prompt secundário –
configurado pela variável
de ambiente PS2

Redirecionamento de Entrada e Saída

Redirecionamento de Entrada Inline



adriana@adriana-VirtualBox: ~

```
adriana@adriana-VirtualBox:~$ wc << OLA
> Primeira frase aula
> Segunda frase AULA
> Final arquivo
> OLA
  3  8 53
adriana@adriana-VirtualBox:~$
```

Cálculos matemáticos

É possível realizar cálculos matemáticos diretamente no bash, e atribuir o resultado de uma operação a uma variável.

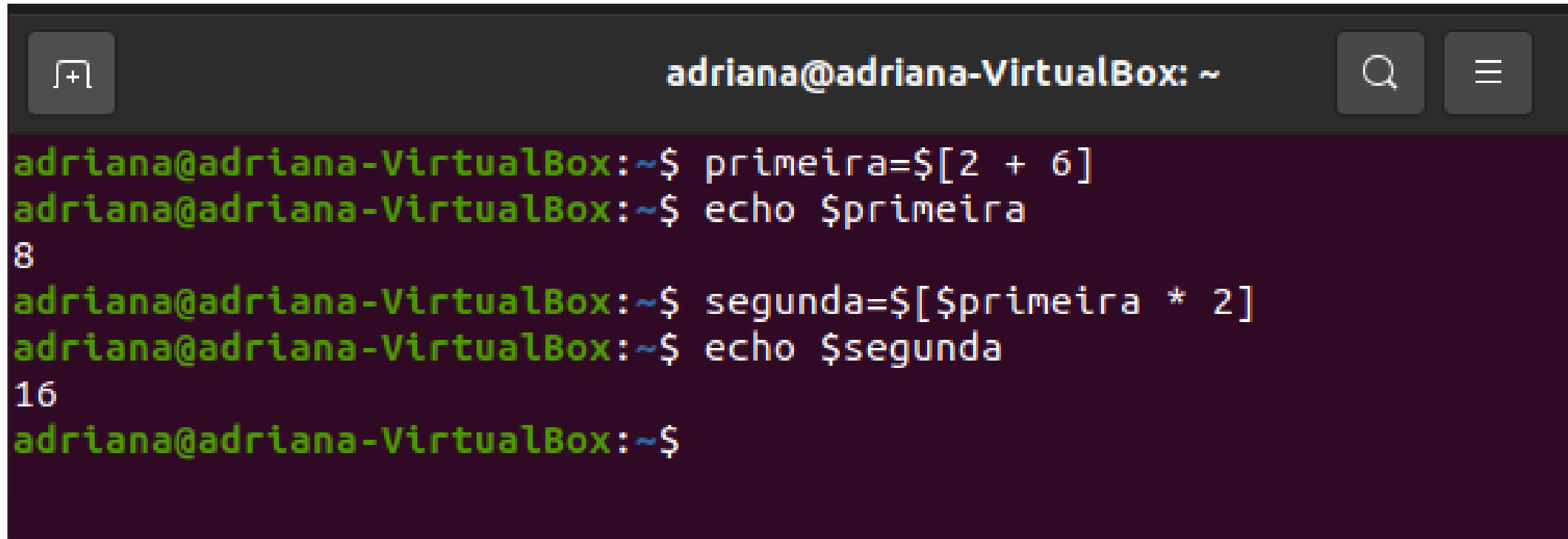
Para isso, englobe a equação usando um cifrão e colchetes:

variável=\${operação}

Exemplo:

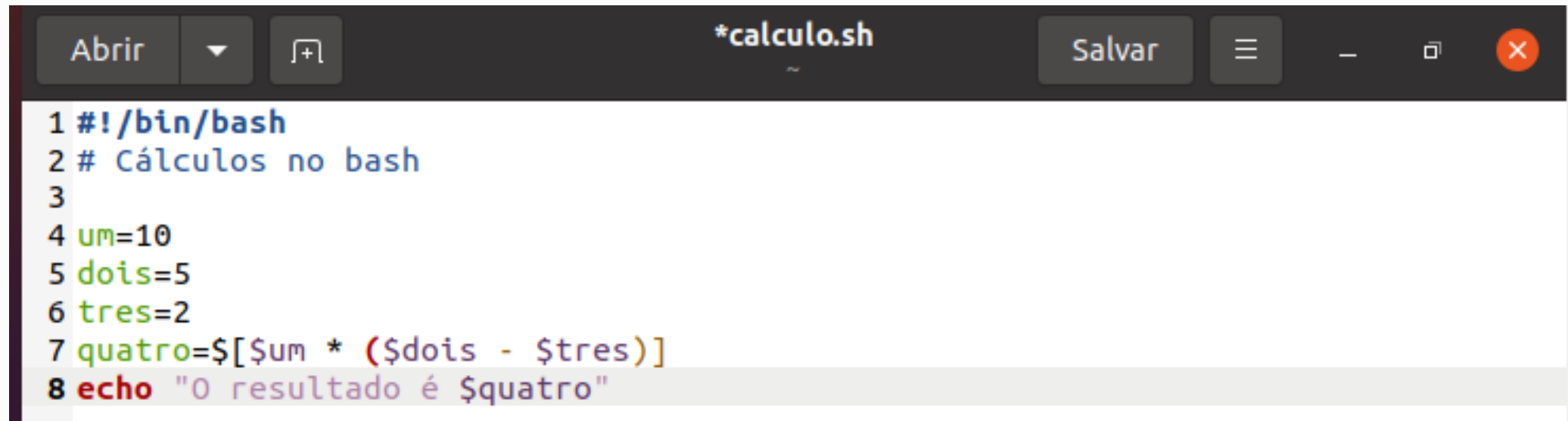
```
primeira=${2+6}  
echo $primeira  
segunda=${$primeira * 2}  
echo $segunda
```

Cálculos matemáticos



```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ primeira=$((2 + 6))  
adriana@adriana-VirtualBox:~$ echo $primeira  
8  
adriana@adriana-VirtualBox:~$ segunda=$((primeira * 2))  
adriana@adriana-VirtualBox:~$ echo $segunda  
16  
adriana@adriana-VirtualBox:~$
```

Cálculos matemáticos



```
1 #!/bin/bash
2 # Cálculos no bash
3
4 um=10
5 dois=5
6 tres=2
7 quatro=$((um * (dois - tres)))
8 echo "O resultado é $quatro"
```

Cálculos matemáticos

```
adriana@adriana-VirtualBox:~$ gedit calculo.sh
adriana@adriana-VirtualBox:~$ ls
  ambteste.sh      Documentos      log.1507212039  Público
'Área de Trabalho' Downloads      meuscript.sh    teste.txt
  calculo.sh       Imagens        Modelos         varusu.sh
  crase.sh         log.1507212038 Música          Vídeos
adriana@adriana-VirtualBox:~$ ls -l
total 64
-rwxrwxr-x 1 adriana adriana 148 jul 14 15:22 ambteste.sh
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 'Área de Trabalho'
-rw-rw-r-- 1 adriana adriana 115 jul 15 21:37 calculo.sh
-rwxr-xr-x 1 adriana adriana 166 jul 15 20:38 crase.sh
drwxr-xr-x 2 adriana adriana 4096 jul  9 11:26 Documentos
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Downloads
drwxr-xr-x 2 adriana adriana 4096 jun 27 14:49 Imagens
-rw-rw-r-- 1 adriana adriana 2460 jul 15 20:38 log.1507212038
-rw-rw-r-- 1 adriana adriana 2524 jul 15 20:39 log.1507212039
-rwxrwxr-x 1 adriana adriana 147 jul 14 15:04 meuscript.sh
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Modelos
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Música
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Público
-rw-rw-r-- 1 adriana adriana 193 jul 15 20:50 teste.txt
-rwxrwxr-x 1 adriana adriana  96 jul 15 20:17 varusu.sh
drwxr-xr-x 2 adriana adriana 4096 jun 11 12:49 Vídeos
adriana@adriana-VirtualBox:~$
```


Cálculos matemáticos

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ chmod 755 calculo.sh  
adriana@adriana-VirtualBox:~$ ls  
ambteste.sh      Documentos      log.1507212039  Público      Vídeos  
'Área de Trabalho' Downloads      meuscript.sh    snap  
calculo.sh       Imagens        Modelos        teste.txt  
crase.sh         log.1507212038 Música         varusu.sh  
adriana@adriana-VirtualBox:~$ ./calculo.sh  
O resultado é 30  
adriana@adriana-VirtualBox:~$
```

Cálculos no bash - problema

Porém, há um problema em realizar cálculos com o shell bash, seus operadores matemáticos só suportam aritmética de números inteiros.

```
var1=9  
var2=2  
var3=$((var1 / var2))  
echo "Resultado: $var3"
```

O resultado deveria ser 4.5, mas o shell retorna 4

Cálculos no bash - problema

```
adriana@adriana-VirtualBox: ~  
adriana@adriana-VirtualBox:~$ chmod 755 calculo.sh  
adriana@adriana-VirtualBox:~$ ls  
ambteste.sh      Documentos      log.1507212039  Público        Vídeos  
'Área de Trabalho' Downloads      meuscript.sh    snap  
calculo.sh       Imagens        Modelos         teste.txt  
crase.sh         log.1507212038 Música          varusu.sh  
adriana@adriana-VirtualBox:~$ ./calculo.sh  
O resultado é 30  
adriana@adriana-VirtualBox:~$ var1=9  
adriana@adriana-VirtualBox:~$ var2=2  
adriana@adriana-VirtualBox:~$ var3=$((var1 / var2))  
adriana@adriana-VirtualBox:~$ echo "Resultado: $var3"  
Resultado: 4  
adriana@adriana-VirtualBox:~$
```

Referências

PRITCHARD, S.; PESSANHA, B. G.; LANGFELDT, N.; STANGER, J.; DEAN, J. 2007. **Certificação Linux LPI Rápido e Prático. Guia de Referência nível 1: Exames 101 e 102.** 2ª Ed. Rio de Janeiro: Editora Alta Books.

Curso de Shell Scripting – Bóson Treinamentos

<http://www.bosontreinamentos.com.br/curso-de-shell-scripting/>