



Introdução à Ciência da Computação – Lista 5

Shell script – parte 2

Nome: Filipe Bessa Carvalho RA: 2024.1.08.009

- 1) No shell, podemos executar uma sequência de comandos, separados por vírgula. Porém, mais vantajoso é encapsulá-los em um arquivo de script. Escolha um editor de texto e crie um arquivo de script, com os comandos `cd /` e `ls -a`, salvando-o como um arquivo com extensão `.sh`. Em seguida, execute-o. Mostre todos os passos para a saída final desse script.

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ gedit cdlsa.sh
ls -la
^C
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls -la
total 12
drwxr-xr-x 2 2024.1.08.009 Domain Users 4096 abr 29 10:15 .
drwxr-xr-x 4 2024.1.08.009 Domain Users 4096 abr 29 10:14 ..
-rw-r--r-- 1 2024.1.08.009 Domain Users 23 abr 29 10:15 cdlsa.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ chmod a+x cdlsa.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls -la
total 12
drwxr-xr-x 2 2024.1.08.009 Domain Users 4096 abr 29 10:15 .
drwxr-xr-x 4 2024.1.08.009 Domain Users 4096 abr 29 10:14 ..
-rwxr-xr-x 1 2024.1.08.009 Domain Users 23 abr 29 10:15 cdlsa.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ cdlsa.sh
cdlsa.sh: command not found
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./cdlsa.sh
. bin cdrom etc lib lib64 lost+found mnt proc run snap swapfile tmp var
.. boot dev home lib32 libx32 media opt root sbin srv sys usr
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

- 2) No exercício anterior, altere o script para exibir uma mensagem “Meu primeiro script”, no final dele. Em seguida, adicione um comando na mesma linha, informando quem está logado no sistema.

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./cdlsa.sh
. bin cdrom etc lib lib64 lost+found mnt proc run snap swapfile tmp var
.. boot dev home lib32 libx32 media opt root sbin srv sys usr
Meu primeiro script
O usuário logado no sistema é: 2024.1.08.009
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

- 3) Crie um script chamado testevariaveis. Declare duas variáveis, uma numérica e uma string, com conteúdo que você desejar. Em seguida, seu script deve exibir uma frase, envolvendo o conteúdo das variáveis criadas. Execute o script.

```
1 #!/bin/bash
2 #Criando e testando variáveis
3 mes=fevereiro
4 dia=15
5 echo "O aniversário do usuário é dia $dia de $mes"
```

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./testevariaveis.sh
O aniversário do usuário é dia 15 de fevereiro
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

- 4) Crie um script chamado testevariaveisambiente. Desenvolva o script de modo que você mostre em tela, em uma frase, o conteúdo de uma variável de ambiente à sua escolha. Procure usar uma variável que não tenha sido mostrada na aula. Execute o script.

```
1 #!/bin/bash
2 #Testando variáveis ambiente
3 echo "O shell utilizado no momento é $SHELL"
```

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls -la
total 20
drwxr-xr-x 2 2024.1.08.009 Domain Users 4096 abr 29 10:39 .
drwxr-xr-x 4 2024.1.08.009 Domain Users 4096 abr 29 10:14 ..
-rwxr-xr-x 1 2024.1.08.009 Domain Users 101 abr 29 10:19 cdlsa.sh
-rwxr-xr-x 1 2024.1.08.009 Domain Users 88 abr 29 10:39 testevariaveisambiente.sh
-rwxr-xr-x 1 2024.1.08.009 Domain Users 118 abr 29 10:35 testevariaveis.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./testevariaveisambiente.sh
O shell utilizado no momento é /bin/bash
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

- 5) Crie um script chamado testecrases. Desenvolva o script de modo que ele mostre inicialmente a data atual. Em seguida o script deve copiar a listagem de diretório (ls -la) para arquivos únicos, diferenciados por data e hora. Execute o script três vezes e mostre o resultado em tela, dos diferentes arquivos.

```

2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./testecrases.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls
cdlsa.sh  log.2904241051.txt  testecrases.sh  testevariaveisambiente.sh  testevariaveis.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./testecrases.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls
cdlsa.sh          log.2904241052.txt  testevariaveisambiente.sh
log.2904241051.txt  testecrases.sh      testevariaveis.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./testecrases.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls
cdlsa.sh          log.2904241052.txt  testecrases.sh          testevariaveis.sh
log.2904241051.txt  log.2904241053.txt  testevariaveisambiente.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ █

```

- 6) Como é feito o redirecionamento do resultado de um comando para um arquivo, de modo que não se sobrescreva o conteúdo do arquivo? Dê um exemplo de como isso pode ser feito, mostrando os resultados. Procure fazer um exemplo diferente do visto na aula.

```

2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./logusuario.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ls
cdlsa.sh  log.txt  logusuario.sh  testecrases.sh  testevariaveisambiente.sh  testevariaveis.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./logusuario.sh
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ cat log.txt
Na data 290424_1100 o usuário era 2024.1.08.009
Na data 290424_1101 o usuário era 2024.1.08.009
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ █

```

```

1 #!/bin/bash
2 data=`date +%d%m%y_%H%M`
3 echo "Na data $data o usuário era $USER" >> log.txt

```

- 7) Como é feito o redirecionamento do conteúdo de um arquivo para ser usado em um comando? Dê um exemplo de como isso pode ser feito, mostrando os resultados. Procure fazer um exemplo diferente do visto na aula.

```

2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ cat < meimprima.txt
"Esse arquivo foi usado como entrada para um comando"
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$

```

- 8) O que consiste o redirecionamento de entrada inline? Dê um exemplo de como isso pode ser feito, mostrando os resultados. Procure fazer um exemplo diferente do visto na aula.

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ wc << MARCA
> 0 marcador delimita o conteúdo
> que será repassado para o comando
> ao ser colocado no início e no final.
> MARCA
      3      19     106
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

- 9) Crie um script chamado scriptaritmético, com uma operação aritmética arbitrária usando pelo menos 4 variáveis, realizando uma operação de divisão cujo resultado não seja um número inteiro. Execute o script e mostre o resultado.

```
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$ ./calcula.sh
0 resultado é 3
2024.1.08.009@suporte-OptiPlex-3050:~/intcc/scripts$
```

```
1 #!/bin/bash
2 um=10
3 dois=3
4 tres=0
5 quatro=$(( $um + $tres ) / $dois ]
6 echo "0 resultado é $quatro"
```