



Introdução à Ciência da Computação – Lista 4 Shell script – parte 1

Nome: Pedro Augusto de Souza Finochio

RA: 2024.1.08.020

1) O que é Shell? O que é Shell Scripting? Dê exemplos de shell existentes no padrão Unix: O shell é um software que expõe serviços e recursos do sistema operacional para um outro usuário ou outro programa. Os shells podem ser de 2 tipos: CLI (Command Line Interface) e GUI (Graphical User Interface). O shell script é uma linguagem de script (conjunto de instruções para que uma função seja executada em determinado aplicativo) usada em ambiente de linha de comando, que permite automatizar comandos de terminal baseados em sistemas baseados em Unix, como Linux e MacOS. Alguns exemplos de shell no padrão Unix são: Bourne Shell (sh): O shell original do Unix, Bourne-Again Shell (bash): Uma extensão popular e moderna do Bourne Shell, amplamente utilizada em sistemas Unix-like, incluindo Linux e o C Shell (csh): Uma shell com uma sintaxe semelhante à linguagem C.

2) O que são variáveis de ambiente? Qual a diferença de variáveis globais e locais? Variáveis de ambiente são variáveis disponíveis ao seu programa/app dinamicamente durante o tempo de execução, são definidas fora de um programa específico e são úteis para armazenar configurações globais, como configurações de linguagem e diretórios padrão. As diferenças entre variáveis globais e locais são que as variáveis globais são definidas dentro de um programa ou script, são acessíveis de qualquer lugar do programa ou script e são usadas para armazenar valores que precisam ser acessados em várias partes do código; Já as variáveis locais são definidas dentro de uma função ou um bloco de código específico, só são acessíveis dentro do escopo especificado e são utilizadas para armazenar valores temporários ou para evitar conflitos de nomes com variáveis globais.

3) Para que servem os comandos env/printenv? Execute-os e mostre o resultado: O comando env pode ser usado para imprimir uma lista das variáveis de ambiente atuais ou para executar outro programa em um ambiente personalizado sem modificar o atual e o comando printenv exibe as informações sobre as variáveis de ambiente.

```
2024.1.08.020@suporte-OptiPlex-3050: $ env
SHELL=/bin/bash
SESSION_MANAGER=local/suporte-OptiPlex-3050:/tmp/.ICE-unix/2423,unix/suporte-OptiPlex-3050:/tmp/.ICE-unix/2423
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
SSH_AGENT_LAUNCHER=gnome-keyring
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
MATHEMATICA_HOME=/usr/share/Wolfram/Documentation/13.3/en-us
LC_ADDRESS=pt_BR.UTF-8
JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64
PAYARA_HOME=/opt/payara6/glassfish
GNOME_SHELL_SESSION_MODE=ubuntu
LC_NAME=pt_BR.UTF-8
SSH_AUTH_SOCK=/run/user/54316/keyring/ssh
IDEA_HOME=/opt/idea-IU-232.8660.185
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
LC_MONETARY=pt_BR.UTF-8
GTK_MODULES=gall:atk-bridge
PWD=/home/2024.1.08.020
LOGNAME=2024.1.08.020
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=wayland
SYSTEMD_EXEC_PID=2468
XAUTHORITY=/run/user/54316/.mutter-Xwaylandauth.LL9PM2
HOME=/home/2024.1.08.020
USERNAME=2024.1.08.020
```

```
2024.1.08.020@suporte-OptiPlex-3050:~$ printenv
SHELL=/bin/bash
SESSION_MANAGER=local/suporte-OptiPlex-3050:@/tmp/.ICE-unix/2423,unix/suporte-OptiPlex-3050:/tmp/.ICE-unix/2423
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
SSH_AGENT_LAUNCHER=gnome-keyring
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
MATHEMATICA_HOME=/usr/share/Wolfram/Documentation/13.3/en-us
LC_ADDRESS=pt_BR.UTF-8
JAVA_HOME=/usr/lib/jvm/java-17-openjdk-amd64
PAYARA_HOME=/opt/payara6/glassfish
GNOME_SHELL_SESSION_MODE=ubuntu
LC_NAME=pt_BR.UTF-8
SSH_AUTH_SOCK=/run/user/54316/keyring/ssh
IDEA_HOME=/opt/idea-IU-232.8660.105
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
LC_MONETARY=pt_BR.UTF-8
GTK_MODULES=gall:atk-bridge
PWD=/home/2024.1.08.020
```

4) **Dê um exemplo de uma variável de ambiente (shell) e seu significado:** Um exemplo de variável de ambiente é a variável de ambiente do sistema que reside na raiz mais alta do sistema e são as mesmas para todo os processos em execução em um sistema sob todos os perfis dos usuários do sistema. Uma dessas variáveis mais comuns é a **PATH** que lista os diretórios onde o sistema operacional deve procurar por executáveis quando você digita um comando no terminal.

5) **Crie uma variável nova no shell atual chamada teste, com o conteúdo “teste”. Mostre em tela o valor da variável criada. Verifique se ela está disponível fora do escopo do shell. Agora faça com que ela seja uma variável global e mostre ela fora do escopo do shell:**

```
2024.1.08.020@suporte-OptiPlex-3050:~$ teste=5
2024.1.08.020@suporte-OptiPlex-3050:~$ bash -c 'if [ -z "$teste" ]; then echo "A variável não está disponível."; else echo "A variável está disponível."; fi'
A variável não está disponível.
2024.1.08.020@suporte-OptiPlex-3050:~$ export teste
2024.1.08.020@suporte-OptiPlex-3050:~$ bash -c 'if [ -z "$teste" ]; then echo "A variável não está disponível."; else echo "A variável está disponível."; fi'
A variável está disponível.
2024.1.08.020@suporte-OptiPlex-3050:~$ █
```

6) **Abra uma nova sessão de shell a partir do mesmo terminal e verifique através do comando ps a existência das duas instâncias de shell rodando ao mesmo tempo. Mostre o valor da variável criada no exercício 5. Em seguida, saia desse shell criado e volte para o anterior. Exclua a variável e mostre que ela foi excluída:**

```
2024.1.08.020@suporte-OptiPlex-3050:~$ bash
2024.1.08.020@suporte-OptiPlex-3050:~$ ps
  PID TTY          TIME CMD
 4908 pts/0        00:00:00 bash
 5789 pts/0        00:00:00 bash
 5804 pts/0        00:00:00 bash
 5811 pts/0        00:00:00 ps
2024.1.08.020@suporte-OptiPlex-3050:~$ echo $teste
5
2024.1.08.020@suporte-OptiPlex-3050:~$ exit
exit
2024.1.08.020@suporte-OptiPlex-3050:~$ unset teste
2024.1.08.020@suporte-OptiPlex-3050:~$ echo $teste

2024.1.08.020@suporte-OptiPlex-3050:~$
```

77) Arrays de ambiente armazenam múltiplos valores dentro de si. Declare um array de ambiente de nome vegetais e preencha com 4 valores (batata, cenoura, beterraba, inhame). Em seguida, mostre em tela o conteúdo total do array. Depois mostre o conteúdo "cenoura" em tela:

```
2024.1.08.020@suporte-OptiPlex-3050:~$ vegetaisarray=( batata cenoura beterraba inhame)
2024.1.08.020@suporte-OptiPlex-3050:~$ echo ${vegetaisarray[*]}
batata cenoura beterraba inhame
2024.1.08.020@suporte-OptiPlex-3050:~$ echo ${vegetaisarray[1]}
cenoura
2024.1.08.020@suporte-OptiPlex-3050:~$
```

8) No array de ambiente do exercício anterior, altere o valor de cenoura para chuchu. Em seguida, remova batata. Em seguida, adicione abobrinha ao array. Em seguida, exclua todo o array. Em todas as operações, mostre em tela o array para ver se as operações foram efetivadas:

```
2024.1.08.020@suporte-OptiPlex-3050:~$ vegetaisarray=(batata cenoura beterraba inhame)
2024.1.08.020@suporte-OptiPlex-3050:~$ vegetaisarray[1]=chuchu
2024.1.08.020@suporte-OptiPlex-3050:~$ echo ${vegetaisarray[*]}
batata chuchu beterraba inhame
2024.1.08.020@suporte-OptiPlex-3050:~$ unset vegetaisarray[0]
2024.1.08.020@suporte-OptiPlex-3050:~$ echo ${vegetaisarray[*]}
chuchu beterraba inhame
2024.1.08.020@suporte-OptiPlex-3050:~$ vegetaisarray[4]=abobrinha
2024.1.08.020@suporte-OptiPlex-3050:~$ echo ${vegetaisarray[*]}
chuchu beterraba inhame abobrinha
2024.1.08.020@suporte-OptiPlex-3050:~$
```

9) Crie aliases para: comando que lista o que contém no diretório do seu home, comando que permite mostrar o conteúdo de um arquivo aos poucos e comando de criação de diretório. Faça testes mostrando que os aliases funcionaram:

```
2024.1.08.020@suporte-OptiPlex-3050:~$ alias lsh="ls -a ~"
2024.1.08.020@suporte-OptiPlex-3050:~$ lsh
.                help                .pki
..               intcc               .profile
AEDS1            intccbackup         Public
arquivo1.txt     .java                saidaBD.txt
.bash_history    .lesshist           snap
.bash_logout     .local              .ssh
.bashrc          manipulaçãodeindicesdeumvetor Templates
BD.txt           .Mathematica        textos
.cache           Music          trabalho
.config          .netbeans            valoresentre100e200
Desktop          NetBeansProjects    Videos
Documents        novo_sequencia.txt   .viminfo
Downloads        objetoofstream       .Wolfram
.gnupg           Pictures
2024.1.08.020@suporte-OptiPlex-3050:~$
```

```
2024-1.08.02@suporte-OptiPlex-3050: $ alias mo="more"
2024-1.08.02@suporte-OptiPlex-3050: $ touch EOUSOUOIMILIOR.txt
2024-1.08.02@suporte-OptiPlex-3050: $ echo "...CRISTIANO DE BICICLETA..MINHA NOSSA, MINHA NOSSA...G0000000000000000000LLLLLLLLLL... RECEBA OS APLAUSOS DO TORCEDOR RIVAL, ESTÁ DE PÉ O TORCEDOR EM TURIN PARA APLAUDIR, PARA REVERENCIAR ESSA MAQUINA, ESSE MONSTRO, ESSE JOGADOR INEXPLICÁVEL CHAMADO CRISTIANO RONALDO..." > EOUSOUOIMILIOR.txt
2024-1.08.02@suporte-OptiPlex-3050: $ no EOUSOUOIMILIOR.txt
...CRISTIANO DE BICICLETA..MINHA NOSSA, MINHA NOSSA...G0000000000000000000LLLLLLLLLL... RECEBA OS APLAUSOS DO TORCEDOR RIVAL, ESTÁ DE PÉ O TORCEDOR EM TURIN PARA APLAUDIR, PARA REVERENCIAR ESSA MAQUINA, ESSE MONSTRO, ESSE JOGADOR INEXPLICÁVEL CHAMADO CRISTIANO RONALDO...
2024-1.08.02@suporte-OptiPlex-3050: $
```

```

2024.1.08.020@suporte-OptiPlex-3050:~$ alias mkd="mkdir"
2024.1.08.020@suporte-OptiPlex-3050:~$ mkd diretório
2024.1.08.020@suporte-OptiPlex-3050:~$ ls -l
total 148
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 17:45 AEDS1
-rw-r--r--  1 2024.1.08.020 Domain Users   35 abr  8 11:33 arquivo1.txt
-rw-r--r--  1 2024.1.08.020 Domain Users  207 abr  8 12:10 BD.txt
-rw-r--r--  1 2024.1.08.020 Domain Users  276 abr 22 16:26 Bike.txt
-rw-r--r--  1 2024.1.08.020 Domain Users    0 abr 22 16:23 CR7.txt
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 16:11 Desktop
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 abr 22 16:32 diretório
drwxr-xr-x  3 2024.1.08.020 Domain Users 4096 abr  1 15:58 Documents
drwxr-xr-x 15 2024.1.08.020 Domain Users 4096 abr 17 13:13 Downloads
-rw-r--r--  1 2024.1.08.020 Domain Users  276 abr 22 16:29 EUSOUOMILIOR.txt
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 abr  8 11:09 help
drwxr-xr-x  3 2024.1.08.020 Domain Users 4096 abr  8 11:09 intcc
drwxr-xr-x  3 2024.1.08.020 Domain Users 4096 mar 27 12:45 intccbackup
-rw-r--r--  1 2024.1.08.020 Domain Users 16905 abr 16 17:44 manipulaçãodeindicesdeumvetor
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 16:11 Music
drwxr-xr-x 16 2024.1.08.020 Domain Users 4096 abr 15 16:20 NetBeansProjects
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 mar 27 13:24 novo_sequencia.txt
-rw-r--r--  1 2024.1.08.020 Domain Users 16886 abr  8 17:54 objetoofstream
drwxr-xr-x  3 2024.1.08.020 Domain Users 4096 mar 13 13:59 Pictures
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 16:11 Public
-rw-r--r--  1 2024.1.08.020 Domain Users    0 abr  8 12:11 saidaBD.txt
drwx----- 5 2024.1.08.020 Domain Users 4096 abr  1 11:40 snap
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 16:11 Templates
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 mar 27 13:13 textos
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 mar 25 13:54 trabalho
-rw-r--r--  1 2024.1.08.020 Domain Users 16909 abr 15 17:35 valoresentre100e200
drwxr-xr-x  2 2024.1.08.020 Domain Users 4096 fev 27 16:11 Videos
2024.1.08.020@suporte-OptiPlex-3050:~$ █

```

10) Crie uma função chamada `primeirafuncao`, que cria um arquivo dentro do seu home, e escreve no arquivo "Olá, Mundo!". Faça isso de modo que toda vez que for executado o conteúdo seja anexado e não substituído:

```

2024.1.08.020@suporte-OptiPlex-3050:~$ primeirafuncao () {
> echo "Olá, Mundo!" >> ~/funcao.txt
> }
2024.1.08.020@suporte-OptiPlex-3050:~$ primeirafuncao
2024.1.08.020@suporte-OptiPlex-3050:~$ ls ~/funcao.txt
/home/2024.1.08.020/funcao.txt
2024.1.08.020@suporte-OptiPlex-3050:~$ cat ~/funcao.txt
Olá, Mundo!
2024.1.08.020@suporte-OptiPlex-3050:~$

```


