Introdução à Ciência da Computação

Shell Script – parte I

Professor Iago Augusto de Carvalho

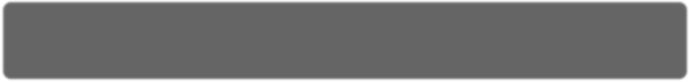
iago.carvalho@unifal-mg.edu.br

1

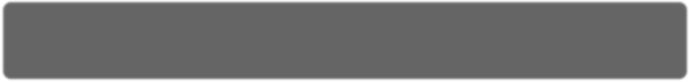
O que é Shell ?

O shell é um programa de usuário, que oferece uma interface personalizável para seus sistemas.

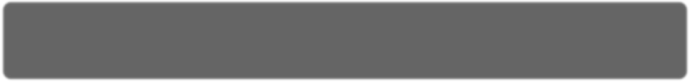
Além disso, os shells:

São uma interface de texto e interativa para o sistema operacional – 

“intérprete de comandos”

São um ambiente de execução de outros programas, consistindo em 

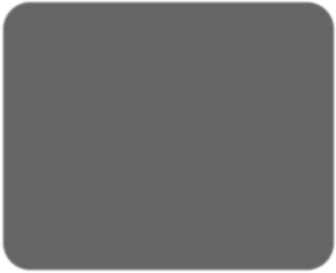
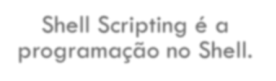
um número de variáveis de ambiente.

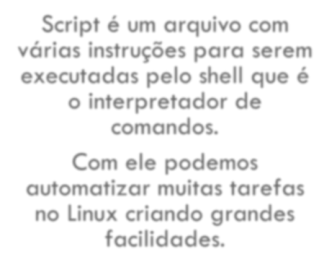
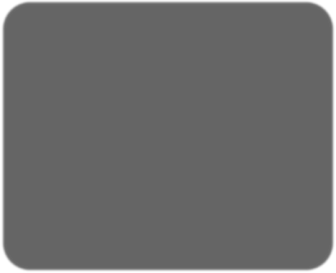
Oferecem suas linguagens de programação. Programas completos 

podem ser escritos nas linguagens dos shells.

2

O que é Shell Scripting?

Shell Scripting é a programação no Shell. 

Script é um arquivo com várias instruções para serem executadas pelo shell que é o interpretador de 

comandos.

Com ele podemos

automatizar muitas tarefas no Linux criando grandes facilidades.

3

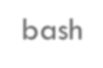
Conceitos básicos

Os shells são personalizáveis, podendo-se ajustar o ambiente operacional. São distintos do kernel.

Executados para uso interativo pelos usuários, e para uso não interativo pelo sistema. Usam vários arquivos de configuração para seus ambientes operacionais. Podem ser executados dentro de outros shells.

4

Exemplos de Shells

sh bash ksh zsh csh tcsh

Você pode instalar shells extras usando: sudo apt install nome\_shell

Ex.: csh, ksh

Shell 

padrão é o **bash**

5

Instalando o csh

6

csh



7

csh



Ctrl + L não funciona

8

sh

9

sh



Ctrl + L não funciona

10

sh, bash e csh



11

Variáveis de Ambiente

Vários programas precisam de informações sobre o usuário e suas preferências para poderem rodar.

Para evitar que você tenha de passar esses dados a cada comando que executa (ex.: caminhos de diretórios de bibliotecas), o shell cria um ambiente operacional composto de inúmeras variáveis.

Essas variáveis contém valores que são usados pelos programas e por outros shells.

Há dois tipos de variáveis

de ambiente:

Variáveis de ambiente

globais Variáveis do shell

(locais)

12

Variáveis de Ambiente

Variáveis globais: podem ser passadas a todos os subprocessos do shell, incluindo outros shells. Exemplos:

PATH – Lista de diretórios de programas executáveis

USERNAME – Nome do usuário logado

TERM – Tipo de terminal ou janela de terminal em uso

HOME – Diretório home do usuário atual

RANDOM – Gera um número aleatório

LANG – Idioma, especificando como locale

Os comandos env e printenv mostram as variáveis de ambiente no terminal. Para criar uma variável global, basta criar uma local e depois exportar com o comando export.

13

Comando 

env

14

Comando 

printenv

15

Variáveis do Shell

São como variáveis locais, pois são específicas do shell atual. Outros programas e shells não as herdam.

Exemplos:

SECONDS – número de segundos desde que o shell foi iniciado.

SHELL – Indica qual o shell em uso atualmente.

16

Variáveis do Shell

17

Variáveis do Shell

Podemos criar uma variável nova digitando um par NOME=valor no terminal : **aula=valor**

E verificamos seu valor com o comando echo:

**echo $aula**

****18

Variáveis do Shell

Podemos disponibilizar a variável criada para outros shells ou programas com o comando export: **export aula**

****

E podemos remover uma variável de ambiente com o comando unset: **unset aula**

****

19

Exemplo

**aula=‘icc’** cria uma variável local e atribui valor **echo $aula** mostra o conteúdo da variável criada 

**bash** cria um novo processo bash filho **ps** mostra os dois processos 

**echo $aula** mostra o conteúdo da variável criada **exit** volta para o processo pai

20

Variáveis locais e subprocessos

**ps** mostra o processo único rodando agora 

**echo $aula** mostra o conteúdo da variável criada

**export aula** exporta a variável AULA

21

Variáveis locais e subprocessos

**bash** cria um novo processo bash filho

**ps** mostra os dois processos

**echo $aula** mostra o conteúdo da variável criada

22

Arrays de Ambiente

Variáveis de ambiente podem ser usadas com arrays – variáveis que podem armazenar múltiplos valores.

Para configurar um array, liste os seus valores entre parênteses:

**frutasarray=(laranja morango banana abacaxi)**

****23

Arrays de Ambiente

Para acessar um elemento no array, use seu número de índice (entre colchetes) contando a partir de zero e coloque todo o conteúdo entre chaves: **echo ${frutasarray[1]}**

****Para mostrar todo o conteúdo do array, use o asterisco \* no lugar do índice: **echo ${frutasarray[\*]}**

****24

Arrays de Ambiente

Alterar valores de uma posição específica: **frutasarray[3]=melancia**

****

Remover itens individuais: **unset frutasarray[0]**

****

Remover o array inteiro: **unset frutasarray[\*]**

****

25

Arrays de Ambiente

Adicionar um novo item: **frutasarray[4]=uva**

****

**Outro exemplo:**

****26

Aliases

É possível definir um alias para um comando ou sequência de comandos usados com frequência.

Para listar os aliases criados, digite alias sem parâmetros. Por padrão, os aliases são locais ao shell e não são repassados a outros programas.

Os alias mostrados aqui, são criados toda vez que o usuário inicializa o sistema. Estão dentro de arquivos de configuração.

27

Aliases

Mostrando os alias que vem definidos

Arquivos de 

configuração

do bash

Para sair do arquivo **:q**

28

Aliases

Mostrando os aliases que vem definidos

29

Aliases

Comando more: Permite fazer a paginação de arquivos ou da entrada padrão. 

O comando more pode ser usado como comando para leitura de arquivos que ocupem mais de uma tela. Quando toda a tela é ocupada, o more efetua uma pausa e permite que você pressione enter ou espaço para continuar avançando no arquivo sendo visualizado. Para sair do more pressione q

30

Aliases

É possível definir um alias para um comando ou sequência de comandos usados com frequência. **alias mo=‘more’**

**alias lshome=‘cd/home;ls’**

31

Aliases 32

Aliases

Também permite modificar o comportamento padrão de um comando.

**alias lsl=‘ls -l’**

****33

Aliases

Também permite modificar o comportamento padrão de um comando.

**alias lsl=‘ls -l’**

****34

Aliases

Para excluir um alias criado, use **unalias nome\_alias**

****

35

Aliases

Para excluir um alias criado, use **unalias nome\_alias**

****36

Funções

O bash também permite trabalhar com funções, que são parecidas com os aliases – mas em vez de comandos, temos pequenos programas.

Sintaxe:

**nome\_função() {comandos}**

Exemplo:

**frase() {**

>**cd /home/adriana**

>**echo ‘Nova Linha’ >> func\_aula**

>**}**

37

Funções

Comando **typeset –f** lista as funções

38

Funções

Para executar a função basta digitar seu nome





39

Funções

Para excluir a função **unset –f nome\_função**

****40

Arquivos de Configuração

Variáveis de ambiente, funções e aliases podem ser configurados automaticamente pelo bash com o uso de alguns arquivos de configuração, de modo a definir o ambiente operacional ao iniciar o sistema.

Cada arquivo de configuração tem um escopo e momento de uso definidos, e o arquivo usado depende do método usado para iniciar o bash.

Há três formas de 

se iniciar o bash:

Como um shell de login padrão na inicialização; 

Como um shell 

interativo que não é o shell de login;

Como um shell não interativo para rodar um script. 

41

Arquivos de 

Configuração

42 Figura: Bóson Treinamentos

Shell de Login

Quando logamos no sistema Linux, o shell bash é iniciado em um shell de login.

Esse shell procura por quatro arquivos de inicialização para processar seus comandos, na seguinte ordem:

/etc/profile ~/.bash\_profile ~/.bash\_login ~/.profile

43

Shell de Login

/etc/profile ~/.profile





44

Shell Interativo

Quando o bash é iniciado como um shell interativo, ele não processa o arquivo **/etc/profile**

Ele tenta executar ao arquivo **.bashrc** no diretório do usuário atual.

O arquivo **.bashrc** pode ser usado para que o usuário insira aliases personalizados e funções de scripts pessoais.

Também executa o arquivo global **/etc/bashrc** ou **/etc/bash.bashrc**, o qual configura algumas variáveis de ambiente locais.

45

Shell Interativo





46

Shell Não Interativo

É o shell iniciado para executar um script (shell script).

Quando um shell é iniciado em modo não interativo, ele verifica a variável de ambiente BASH\_ENV para descobrir se há comandos de inicialização a executar.

Por padrão, o valor desta variável não é configurado.



47

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/

48