

Bash

Tiago H31nr1ch
heinrichtx@gmail.com

03/2020

1 Calendário de Atividades

- A nota D1 será composta dos exercícios de \LaTeX e a atividade na Seção 3. A atividade deve ser entregue até o dia 10/04 e também vai contar presença para a aula do dia 02/04.
- A atividade que foi realizada nas últimas aulas (Linux: Activity .pdf file.) deve ser entregue até o dia 05/04, esta nota irá completar a nota D2.

2 Estruturas em Bash

O objetivo desta aula é apresentar operações e estruturas as quais possam ser aplicadas em um conjunto para solucionar rotinas repetitivas.

2.1 Loops

O famoso *while* **True** acaba sendo utilizado em repetição de rotinas e até mesmo em verificação de *status*.

```
1 #!/bin/bash
2 while true; do
3     echo ok
4 done
5
6 until false; do
7     echo ok
8 done
```

2.2 Funções

Para acessar os argumentos dentro da função é utilizado \$1, \$2, \$3 e assim por diante. Caso seja necessário realizar alguma alteração no valor de entrada (como \$1), é recomendado utilizar uma variável local. Para referenciar todos os argumentos pode ser utilizado “\$@”.

```
1 #!/bin/bash
2 function basic_input () {
3     echo $1 $2
4     local value="$3"
5     echo "random: $value"
6 }
7
8 basic_input 1 2 3
```

Argumentos padrão;

```
1 #!/bin/bash
2 foo() {
3     local value=${1:-100}
4     local value1=${2:-30}
5     echo "$value $value1"
6 }
```

Requer um argumento:

```
1 foo1() {
2     local value=${1:?Argument}
3     echo "$value"
4 }
```

Para verificar uma função, basta utilizar *declare -f nome_funcao*.

A instrução de retorno no Bash não retorna um valor como uma função em C (só existe um retorno de status). Você pode pensar nisso como o status de saída dessa função. Caso é deseja o retorno de um valor, uma solução é:

```
1 function foo2(){
2     local value="random string"
3     echo $value
4 }
5
6 var="$(foo2)"
7 echo $var
```

2.3 Find

O *locate* é responsável por efetuar uma indexação do sistema por *hash*. No caso de uma busca “local” existe a possibilidade de utilizar o comando *find*.

```
1 find . -name "myFile.txt"
2 find . -name "*.txt" -o -name "*.sh"
3 find /opt -type f #FILE
4 find /opt -type d #DIR
5
6 find . -mmin -120 #
7 find . -mmin +120
8
9 -mmin -n : 0 arquivo foi modificado há menos de n minutos
10 -mmin +n : 0 arquivo foi modificado há mais de n minutos
11 -mtime
12
13 find -type f -size +15M
```

2.4 Sort

sort é um comando Unix para ordenar dados em arquivos em uma sequência.

```
1 ls -l | awk '{print $3}' | sort -u # unique
2
3 cat out.txt | sort -n #numeric
4 cat out.txt | sort -r #reverse
5
6 ls -l | sort -k 5 -nr #column {5}
```

3 Atividade Nota D1

1. Visando otimizar rotinas no sistema, escreva uma função que tenha o mesmo retorno de *ls -la* e apresente o número de arquivos ocultos no respectivo diretório.
2. Escreva uma função que realiza uma busca no diretório */etc/* , de uma *string* que foi recebida.
3. Faça uma função que recebe o nome de um processo e retorna o *pid* do respectivo.
4. Escreva um script que recebe um valor X ao qual representa o número de entradas que será recebido do usuário. Para cada entrada do usuário você deve realizar a soma com o valor anterior. Por exemplo:

Número de interações:

2

Entrada 1:

10

Entrada 2:

20

Total: 30

5. Ordene o conteúdo do `/etc/` em ordem reversa (pelo nome)
6. Ordene somente arquivos do tipo `.conf` no `/etc/`.
7. Faça uma função de busca que receba o diretório e tipo de arquivo.

Para realizar as atividades é necessário fazer o download do arquivo (**Material D1.**), na máquina virtual. O download pode ser realizado com **wget** <https://h31nr1ch.github.io/data/data-d1.zip> e para descomprimir basta utilizar o comando **zip** **data-d1.zip** (lembre de instalar o zip através do **apt**). Dois arquivos estão presentes no `data.zip`, estes sendo o `config.ini` e `quotes.json`. A atividade consiste da implementação de rotinas para alterar ambos arquivos.

8. No arquivo `quotes.json`, remova todas as “,” e substitua todos os “{” e “}” por “[” e “]” respectivamente.
9. Faça duas funções uma que realize a leitura de todo o arquivo `quotes.json` e imprima na tela somente o nome de todos os autores e outra que só imprima suas respectivas citações (`quotes`).
10. Algumas citações como “*I love deadlines. I like the whooshing sound they...*”, estão divididas por um “.” faça uma função que só imprima a primeira parte da frase (neste caso “*I love deadlines.*”).
11. No arquivo `config.ini` utilize `sed` para substituir o endereço 192.168.0.4 por 200.19.103.210.
12. Faça uma função que identifique todos as linhas `#__run = no` e substitua as opções `no` por `yes`.
13. Imprima na tela todas as portas configuradas no arquivo (como `#__port =` ou `#__port_#` =)
14. Utilizando o `sed` troque o nível do `syslog` de Error para Warning.