



## Introdução à Ciência da Computação – Lista 7

### Shell script – parte 4

Nome: Francisco Toro Tonissi RA: 2025.1.08.025

- 1) Crie um script chamado escrevenome, faça com que a saída desse script seja seu nome completo. Não utilize o comando chmod. Depois crie um script chamado testecompara, utilize o operador AND e verifique se o usuário logado tem permissão r e x sobre o script escrevenome. Mostre o resultado da saída.

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano escrevenome.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash escrevenome.sh
Francisco Toro Tonissi
2025.1.08.025@suporte-OptiPlex-3050:~$
```

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano testecompara.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash testec
testecompara.sh testecrases.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash testec
testecompara.sh testecrases.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash testecompara.sh
O usuário 2025.1.08.025 NÃO tem permissão de leitura e execução sobre ./escrevenome.
2025.1.08.025@suporte-OptiPlex-3050:~$
```

- 2) Crie um script chamado frutascase. Com base no valor da variável fruta mostre uma breve descrição da fruta. Faça com 5 frutas. Exemplo: fruta=uva, echo “A uva é o fruto da videira ou parreira, uma planta da família Vitaceae. É originária da Ásia e uma das frutas mais antigas utilizadas na alimentação humana. Existem mais de 60 mil variedades da fruta. A cor, o sabor e o tamanho variam de acordo com cada espécie. A uva também é classificada quanto ao destino de produção, de mesa ou para vinicultura. Pode ser consumida in natura ou usada na preparação de doce, vinho, passas, musses, geleias, tortas, gelatinas, sucos.”

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano frutas.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ nano frutas.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ nano fruta.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash fruta.sh
A uva é uma fruta da videira, muito antiga, com milhares de variedades. Pode ser consumida in natura, em vinhos, sucos e doces.
2025.1.08.025@suporte-OptiPlex-3050:~$ bash fruta.sh
A uva é uma fruta da videira, muito antiga, com milhares de variedades. Pode ser consumida in natura, em vinhos, sucos e doces.
2025.1.08.025@suporte-OptiPlex-3050:~$ nano fruta.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash fruta.sh
O morango é uma fruta doce-ácida, rica em vitamina C. Muito usado em sobremesas, sucos e geleias.
2025.1.08.025@suporte-OptiPlex-3050:~$ nano fruta.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash fruta.sh
A banana é tropical, rica em potássio e fibras. Muito consumida, serve para doces, vitaminas e pode ser comida crua ou frita.
2025.1.08.025@suporte-OptiPlex-3050:~$ nano fruta.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash fruta.sh
O abacaxi é uma fruta ácida e refrescante, rica em vitamina C e boa para a digestão. Usado em sucos, doces e pratos salgados.
2025.1.08.025@suporte-OptiPlex-3050:~$
```

- 3) Cite, explique e faça um script simples para cada estrutura de repetição do shell bash. Use sua criatividade para os scripts.

For: o for é usado para repetir uma função por um determinado período de tempo.

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano forpratico.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash forpratico.sh
Contando de 1 a 5 com for:
Número: 1
Número: 2
Número: 3
Número: 4
Número: 5
2025.1.08.025@suporte-OptiPlex-3050:~$
```

For pratico:

While: É um loop que executa enquanto uma condição for verdadeira.

While pratico:

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano whilepratico.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash whilepratico.sh
Contagem progressiva com while:
2025.1.08.025@suporte-OptiPlex-3050:~$ nano whilepratico.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash whilepratico.sh
Contagem progressiva com while:
Já foram 15 segundos...
Já foram 16 segundos...
Já foram 17 segundos...
Já foram 18 segundos...
Já foram 19 segundos...
Já foram 20 segundos...
2025.1.08.025@suporte-OptiPlex-3050:~$
```

Until: Funciona como o while, mas só acontece o loop quando for falso

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano untilpratico.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash untilpratico.sh
Digite a senha: 1234
Digite a senha: d2
Digite a senha: ICC 2025
Senha correta!
2025.1.08.025@suporte-OptiPlex-3050:~$
```

until pratico:

4) Explique o que é IFS e faça um script diferente do que foi visto em aula. Use sua criatividade.

IFS é o comando que se usa para ler uma lista de tarefas, fornecidas em uma linha, e ela exibe cada item de forma numerada.

IFS pratico:

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano listatarefas.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash listatarefas.sh
Digite suas tarefas separadas por vírgula (ex: Estudar,Limpar,Comprar pão):
Estudar, treinar, rezar
Suas tarefas são:
1) Estudar
2) treinar
3) rezar
2025.1.08.025@suporte-OptiPlex-3050:~$
```

5) Crie um script for no estilo C que mostre na tela os números de 50 a 20.

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano 50a20estiloC.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash 50a20estiloC.sh
Números de 50 a 20 (estilo C):
50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
20
2025.1.08.025@suporte-OptiPlex-3050:~$
```

- 6) Desenvolva um script que receba um parâmetro e verifique se o valor está entre 0 e 10. Caso sim mostre o triplo do valor. Caso ele esteja entre 10 e 20 mostre o dobro. Caso não esteja nos anteriores apresente uma mensagem.

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano verificavalor.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash verificavalor.sh
Uso: verificavalor.sh <número>
2025.1.08.025@suporte-OptiPlex-3050:~$ bash verificavalor.sh 9
O valor está entre 0 e 10. Triplo: 27
2025.1.08.025@suporte-OptiPlex-3050:~$
```

- 7) Explique o que é \$# e faça um script diferente do que foi visto em aula. Faça com dois parâmetros. Use sua criatividade.

R: é uma **variável especial** no Bash que representa o **número de argumentos** passados para um script na linha de comando.

```
2025.1.08.025@suporte-OptiPlex-3050:~$ nano calculadora.sh
2025.1.08.025@suporte-OptiPlex-3050:~$ bash calculadora.sh
Erro: você deve fornecer exatamente 2 números.
Uso: calculadora.sh <número1> <número2>
2025.1.08.025@suporte-OptiPlex-3050:~$ bash calculadora.sh 4 5
Você digitou: 4 e 5
Soma: 9
Multiplicação: 20
2025.1.08.025@suporte-OptiPlex-3050:~$
```