



1) Sabendo que a nona coluna delimitada por espaço em branco no arquivo `access.log` indica o status do pedido e que um status 404 significa que o arquivo pedido não foi encontrado, determine quantos arquivos foram pedidos e não encontrados no log.

2) Quantos dos arquivos da questão 1 foram pedidos no mês de Junho?

3) Utilizando pipe, crie uma linha de comando que:

* cria uma lista longa com o conteúdo de `/usr/bin`

* mostra somente os tamanhos e nomes dos arquivos encontrados, ordenados por tamanho

4) Faça a mesma coisa da questão 3, porém utilizando um *shell script* que receba o diretório a ser listado como parâmetro.

5) Escreva um shell script que receba exatamente dois números como parâmetro e mostre quem é maior entre eles. Por exemplo:

```
prompt$ ./maior.sh 3 5
5 é maior
prompt$ ./maior.sh 7 6
7 é maior
prompt$ ./maior.sh 5 5
Os números são iguais
```

6) O arquivo `/etc/passwd` mostra o login e o nome completo de cada usuário do sistema (campos 1 e 5) separados por um tab. Seu último campo mostra o shell utilizado por cada usuário. Escreva um script que mostre login, o nome do usuário e o shell utilizado para todos os usuários do sistema. Por exemplo:

```
prompt$ ./users.sh
Login:ftp      Nome:FTP User      Shell: /bin/bash
Login:nobody   Nome:Nobody        Shell: /bin/false
Login:named    Nome:Domain name  server Shell: /bin/bash
Login:305013   Nome:José Silva   Shell: /bin/bash
prompt$
```

7) Escreva um shell script que receba duas palavras como parâmetro e verifique se a primeira palavra está contida dentro da segunda. Só mostre uma mensagem informativa em caso de sucesso; do contrário, não mostre nada. Exemplo:

```
prompt$ ./substring.sh ana banana
ana está contida em banana
prompt$ ./substring.sh banana maria
prompt$ ./substring.sh banana
prompt$ ./substring.sh
prompt$
```

8) Escreva um shell script que conte quantos arquivos um usuário possui em seu diretório e informe esta quantidade. Este script deve ser executado sempre que um terminal for aberto pelo usuário.

9) Escreva um shell script que compacte todo o conteúdo do seu diretório trabalhos. Este script deve apagar os arquivos após a compactação. Este script deve, ainda, ser executado às 23h59 de todas as segundas feiras do mês.

Dica: para compactar arquivos, use o comando `zip <nomedoarquivo.zip> <nomedo(s)arquivo(s)>`

10) Escreva um shell script que, a partir do arquivo `access.log`, que deve ser passado por parâmetro, calcule a quantidade total de bytes transferidas em um mês de acessos ao servidor.

Dica: observe a coluna que conta os bytes transferidos. Use `grep` para selecionar o mês.