

Sistemas Operativo

Concurso 1

Nome: Afonso Rio Soares da Silva

Número: 76943

Q1:

a)

```
#!/bin/bash
read -p 'enter extension: ' ex
ls *.$ex
```

Lemos uma mensagem do terminal enviada pelo user, transformamos a mensagem numa variável “ex”. O comando “ls *.\$ex” vai listar todos os ficheiros que têm a extensão “ex”.

b)

```
#!/bin/bash
read -p 'enter size: ' s
find ./ -size +"$s"c
```

Lemos uma mensagem do terminal enviada pelo user, transformamos a mensagem numa variável “s”. O comando “find” vai procurar em todos os ficheiros (./*) o tamanho (-size) maior que aquele que foi colocado no input (“+\$s”). O c indica que usamos bytes.

c)

```
#!/bin/bash
read -p 'enter first date: ' d1
read -p 'enter second date: ' d2
find -newerct $d1 ! -newerct $d2
```

Lemos duas datas que são pedidas no terminal, sendo elas a “d1”(data 1) e “d2”(data 2). Usamos o comando find para procurar files que são mais velhos que “d1” e os que são mais novos que “d2”, por outras palavras:

```
if(dateOfCurrentFile => d1 && dateOfCurrentFile <= d2)
```

```
print(CurrentFile).
```

d)

```
#!/bin/bash
cut -d: -f1,7 /etc/passwd --output-delimiter=' '
```

Com o comando cut vamos cortar o conteúdo do ficheiro “passwd”(onde existe a informação sobre os users do PC) , usamos o “-d:” para delimitarmos o texto mostrado com os “:” pois no ficheiro “passwd” a informação esta separada por “:”, e como queremos a “coluna” 1 e 7 usando o comando “-f1,7”. O comando “—output-delimiter=’ ’” vai substituir o delimiter “:” por “ ”.

e)

```
#!/bin/bash
read -p "String que deseja procurar: " string
grep -l "$string" ./*
```

Lemos um o input de um texto ou frase do user e depois vamos procurar esse texto com o comando grep que nos permite procurados palavras em todos os files (./*), usamos o comando -l para mostra-mos apenas o nome do file.

Q2:

a)

```
#!/bin/bash
read -p "write a number: " number
input=$number
result=1
while [ $number -gt 1 ]
do
result=$((result*$number))
number=$((number - 1))
done
echo "The factorial of $input is: $result"
```

Vamos ler um número que o user envia no input e guardamos na variável “number”

Depois criamos outra variável “result” com o valor de 1. Em seguida criamos um ciclo while em que a condição é: Se o Number(numero introduzido no input) for maior ou igual a 1 então o ciclo continua, caso essa condição seja quebrada o ciclo acaba. Dentro do ciclo vamos fazer a operação result = result * number, assim vamos multiplicando o result pelos valores de “number” como se fosse o fatorial(number!), e em seguida retiramos o valor 1 á variável “number” . No final se o “number” for 3 imprimimos:

“The factorial of 3 is: 6”

b)

```
#!/bin/bash
read -p "Insert number to do the multiplication table: " number
read -p "Insert the limit of the table: " limit
i=0
while [ $limit -ge $i ]
do
    result=$((i*number))
    echo "$i x $number = $result"
    ((i++))
done
```

Lemos duas variáveis do user a variável “number” que vai ser a tabuada do valor que queremos por exemplo 5, e o limite da tabua, podemos só querer a tabuada do 5 ate 4.

Criamos uma variável i que damos o valor de 0 inicialmente. Em seguida criamos um ciclo while com a condição se o “limite” for maior ou igual a “i” então o ciclo continua se o contrario acontecer o ciclo acaba. Dentro no ciclo temos uma variável result que vai representar o resultado da conta por exemplo se fizermos a tabua do 5 ate 2 o result iria ter o resultado de 0, em seguida de 5 e por ultimo de 2. Depois para representarmos a tabuada no ecrã do user damos um echo com a seguinte string se por exemplo o “i” no momento for 4 e o “number” for 5:

4 x 5 = 20

E por ultimo adicionamos “i” por 1 para avançarmos na tabuada do valor “number” ate ao seu limite(limit)

Q3:

a)

```
#!/bin/bash
read -p "Extension you want to modify: " Ex1
read -p "Extension you want to Replace $Ex1 with: " Ex2
for file in *.$Ex1
do
    mv "$file" "${file%.*}.$Ex2"
done
```

Lemos duas variáveis, “Ex1” a extensão dos ficheiros que queremos modificar e “Ex2” o que queremos substituir pela extensão “Ex1”. Depois usamos um ciclo “for” para percorrermos os files que existem no PC, mas só queremos aqueles com a extensão

Ex1. Dentro do ciclo vamos ter o comando “mv” que vai renomear o ficheiro, \$file o file onde estamos agora no ciclo vai ser renomeado com a extensão “Ex2”.

b)

```
#!/bin/bash
read -p "Enter the name of the first file: " file1
read -p "Enter the name of the second file: " file2
cat $file1 $file2 > file3
wc -l < file3
```

Lemos o nome de dois ficheiros que vão guardados respetivamente nas variáveis “file1”, “file2”.

Em seguida chamamos o comando “cat” e criamos um novo file(file3) que vai juntar o conteúdo do “file1” e “file2”, por ultimo usamos o comando “wc -l < file3” para mostrar as linhas que o “file3” tem.

Q4:

```
#!/bin/bash
read -p "Insert year: " year
i=$((year%4))
j=$((year%100))
k=$((year%400))
if [ $i = 0 ] && [ $j ≠ 0 ]
then
result=29
echo "The number of days in February in the year $year is: $result"
elif [ $i = 0 ] && [ $j = 0 ] && [ $k = 0 ]
then
result=29
echo "The number of days in February in the year $year is: $result"
else
result=28
echo "The number of days in February in the year $year is: $result"
fi
```

Lemos do um número que o user vai colocar no input que vai representar o ano que queremos verificar se é bissexto ou não. Começamos por criar três variáveis “i”, “j” e “k”. Como sabemos um ano só é bissexto se esse ano for divisível por 4 e não por 100 ao mesmo tempo, ou se o ano por divisível por 4, 100 e 400 ao mesmo tempo. Então colocamos um “if” para verificar a primeira condição se o ano é divisível por 4 e não por 100 ao mesmo tempo, se sim imprimimos que fevereiro tem 29 dias, se não verificamos o próximo “if”, se o ano é divisível por 4, 100 e 400 ao mesmo tempo, se sim imprimimos que fevereiro tem 29 dias, se não vamos para o “else”, onde nenhuma das condições foi cumprida então fevereiro tem apenas 28 dias.

