

Universidade do Algarve
Faculdade de Ciências e Tecnologia
Licenciatura em Engenharia Informático

Sistemas Operativos

Relatório 1º Guião Prático



Discente: Hugo Paixão, nº 64514

Docentes: Amine Berqia e José Bastos

Índice

1. Nota prévia	3
2. Introdução	3
2.1 Sistemas Operativos	3
2.2 Shell	3
2.3 Shell scripts	3
2.4 Bash	4
3. Alguns comandos em shell.....	4
4. Programação em Shell	7
4.1 Variáveis e operadores	7
4.1 Estruturas de controlo de execução em bash	9
5. Shell scripts	10
6. Conclusão	15
7. Webgrafia	15

1. Nota prévia

Para além dos scripts estarão também anexados printscreens da sua execução.

2. Introdução

2.1 Sistemas operativos

Sistema operativo funciona como intermediário entre o utilizador do computador e os recursos físicos (hardware). Este tem como principais funções:

- A gestão dos programas em execução - criar e eliminar os programas, suspender e arrancar os programas, ordenar a execução dos programas, fornecer mecanismos para a sincronização e comunicação entre programas.
- A gestão de memória – manter informação sobre que parte da memória estão a ser usados e por quem, decidir a que programas atribuir espaço livre de memória, atribuir e libertar memória de acordo com as necessidades.
- A gestão do sistema de ficheiros – criar e eliminar diretórios e ficheiros, relacionar os documentos com o espaço físico que ocupam, manter a integridade dos dados durante todas as operações, gerir as permissões de acesso aos dados.

2.2 Shell

Shell é uma interface de utilizador para acessar aos serviços de um sistema operacional. Em geral, shells do sistema operativos usam uma interface de linha de comandos ou uma interface gráfica de utilizador, dependendo da função e operação particulares de um computador. É chamado de Shell porque é a camada mais externa em torno do núcleo do sistema operacional.

2.3 Shell script

Um shell script é um ficheiro que contém uma sequência de comandos para sistemas operacionais baseados em Unix. Este contém uma sequência de comandos. Os scripts são geralmente usados quando é necessário usar comandos repetidamente. Como outros programas, os scripts podem conter parâmetros, comentários e subcomandos que o shell deve seguir. Inicia-se a sequência de comandos no script simplesmente inserindo o nome do arquivo na linha de comandos.

Os scripts contêm texto ASCII e são escritos usando um editor de texto. O conteúdo do script é uma série de comandos em uma linguagem que pode ser interpretada pelo shell. As funções que os scripts suportam incluem loops, variáveis, instruções if/then/else, arrays e atalhos. Depois de

concluído o script é guardado normalmente com uma extensão .txt ou .sh e em um local que o shell pode acessar.

2.4 Bash

Bash é um interpretador de comandos entre o utilizador e o sistema operativo conhecido como shell. O acrónimo Bourne-Again SHell, o bash é uma evolução retro compatível muito mais interativa do shell. Este permite a execução de sequências de comandos inseridos diretamente na linha de comandos(“prompt”) ou ainda lidos de arquivos de texto conhecidos como shell scripts. O bash apresenta recursos e características de uma linguagem de programação de alto nível. É compatível por configuração com as normas POSIX, de forma que os scripts Bash podem ser executados em diversos sistemas tipo Unix.

3. Alguns Comandos

Comando	Definição	Exemplo
man	Manual completo acerca de um comando concreto.	\$ man who
info	Abre o explorador de informações.	\$ info who
id	Retorna os identificadores do usuário, login e os grupos a que ele pertence.	\$ id -g
whoami	Retorna o nome do atual utilizador da shell.	\$ whoami
who	Retorna informação acerca do utilizador logado no sistema.	\$ who
tty	Retorna o diretório onde o terminal está inserido.	\$ tty
finger <user>	Mostra informação acerca de determinados utilizadores do sistema.	\$ finger nuno
last	Indica os últimos logins de utilizadores.	\$ last
date	Indica a data e hora exatas.	\$ date
time <command>	Mede o tempo de execução de determinado programa.	\$ time whoami
passwd	Possibilita a alteração da password de certo utilizador.	\$ passwd nuno
echo <argument>	Escreve a expressão passada como argumento.	\$ echo “Bom trabalho!”
mesg	Permite/nega a troca de mensagens de terminal.	\$ mesg y
mail/mailx	Enviar/receber emails.	-
su	Substitui o utilizador.	\$ su
bc	Retorna uma calculadora no terminal.	\$ bc
cal	Retorna um calendário no terminal.	\$ cal

clear	Limpa a janela do terminal.	-
ps	Apresenta informações acerca de um determinado processo.	\$ ps
jobs	Lista os processos que estão a correr.	-
<Ctrl> <z>	Recomeça o terminal.	-
bg <pid>	Resume programas suspensos, correndo-os como programas de segundo plano.	\$ bg 666
bg <job_id>	Resume processos suspensos, correndo-os como programas de segundo plano.	\$ bg 666
Comando	Definição	Exemplo
fg <pid>	Resume programas suspensos, correndo-os como programas de primeiro plano.	\$ fg 666
fg <job_id>	Resume processos suspensos, correndo-os como programas de primeiro plano.	\$ fg 666
pwd	Retorna o caminho desde a raiz até ao diretório atual.	\$ pwd
ls	Lista o conteúdo dos diretórios designados.	\$ ls
cd	Altera o diretório de trabalho atual para outro designado.	-
find	Retorna uma hierarquia de ficheiros.	\$ find SO_1920
locate	Localiza e retorna a localização de um ficheiro pelo nome.	\$ locate SO_1920
mkdir	Cria diretórios.	\$ mkdir SO_2021
rmdir	Remove diretórios vazios do sistema.	\$ rmdir SO_1920
chmod	Altera o tipo de acesso a um determinado ficheiro.	\$ chmod +x FIFA19.exe
cat	Altera o tipo de acesso a um determinado ficheiro.	\$ cat 1.txt 2.txt > 3.txt
more	Mostra um ecrã de cada vez, se um ficheiro é grande.	\$ more +3 1.txt
less	Mostra uma página de cada vez, se um ficheiro é grande.	\$ less 1.txt
cut	Corta secções de cada linha de um certo ficheiro e retorna o resultado.	\$ cut -d' ' -f3,7,19 1.txt
vi	Executa o editor visual do sistema operativo.	\$ vi amtlb.dll
head <filename>	Retorna os primeiros N elementos de um determinado ficheiro.	-
tail <filename>	Retorna os últimos N elementos de um determinado ficheiro.	-
ln	Cria ligações entre ficheiros.	\$ ln -s /ano2/SO_1920
cp	Copia ficheiros ou diretórios.	\$ cp amtlb.dll
mv	Move e renomeia ficheiros ou diretórios.	\$ mv ano1 ano_1

rm	Elimina ficheiros ou diretórios.	\$ rm teste.txt
file	Define o tipo (de leitura) de um determinado ficheiro.	\$ file Imagens
wc	Conta o número de algo (linhas, palavras, bytes, char, etc.) num ficheiro.	\$ wc teste.txt
split	Divide um ficheiro em partes.	-

Comando	Definição	Exemplo
grep <expressão><ficheiro>	Retorna todas as linhas que contenham uma determinada expressão num certo ficheiro.	\$ grep feito cadeiras.txt
sort	Ordena um ficheiro por uma determinada ordem.	\$ sort teste.txt
diff	Analisa dois ficheiros, simultaneamente, e retorna as linhas diferentes.	-
compress	Reduz o tamanho de um determinado ficheiro.	-
uncompress	Descomprime ficheiros previamente comprimidos.	-
chgrp	Altera o grupo que detém um determinado ficheiro ou diretório.	-
chown	Altera o grupo e/ou o utilizador que detém um determinado ficheiro ou diretório.	-
df	Apresenta informação relacionada com o espaço total/disponível do sistema.	\$ df
du	Apresenta a porção de memória do sistema que um determinado ficheiro ocupa.	\$ du
mount	Organiza os ficheiros de sistema de um certo dispositivo em forma de árvore.	\$ mount -t iso666 mnt/cdrom
unmount	"Desmonta" ficheiros previamente organizados em forma de árvore.	\$ unmount mnt/cdrom

Tabela 1 – Comandos elementares, definição e exemplos singulares.

Umask	Arquivo		Diretório
	Binário	Texto	
0	r-x	rw-	rwX
1	r--	rw-	rw-
2	r-x	r--	r-X

3	r--	r--	r--
4	--X	-w-	-wX
5	---	-w-	-w-
6	--X	---	--X
7	---	---	---

Tabela 2 – Umask a utilizar.

4. Programação em Shell

4.1 Variáveis e Operadores

Parâmetros especiais:

Representa o número de argumentos passados ao programa: \$#

Referencia todos os parâmetros posicionais: \$*

Representa o nome do programa ou script a ser executado: \$0

Representa o número do processo a ser executado: \$\$

Representa o número do último processo executado em segundo plano: \$!

Representa o estado do último comando não executado em segundo plano: \$?

Operadores lógicos:

AND: -a / \$\$

OR: -o / ||

NOT: !

Operadores relacionais:

Menor do que: -lt

Menor ou igual a: -le

Igual a: -eq

Maior do que: -gt

Diferente: -ne

Operadores de manipulação de caracteres (strings):

Iguais: =

Diferentes: !=

De comprimento maior que zero (não nula): -n

De comprimento igual a zero (nula): -z

Operadores de manipulação de ficheiros:

É um diretório: -d

É um ficheiro regular: -f

É um ficheiro com permissão de leitura: -r

É um ficheiro com permissão de escrita: -w

É um ficheiro com permissão de execução: -x

É um ficheiro com conteúdo: -s

É um ficheiro do utilizador atual: -O

É um ficheiro do grupo atual: -G

É um pipe: -p

Mais recente: -nt

Mais antigo: -ot

4.2 Estruturas de controlo de execução em bash

Estrutura	Sintaxe	Exemplo
if	if [<condição1>]; then	echo Passou no exame?

	<pre><comandos> else <comandos> fi</pre>	<pre>read resposta if [\$resposta = "sim"]; then echo "Parabens :)" else echo "Não estudou :(" fi</pre>
for	<pre>for ((exp1;exp2;exp3)) do <comandos> Repete todos os comandos enquanto exp2 seja verdadeiro done</pre>	<pre>for ((i = 0; i<= 9; i++)) do echo \$i done</pre>
case	<pre>case "\$<variável>" in "\$<condição1>") <comandos>; "\$<condição2>") <comandos> ;; esac</pre>	<pre>echo "Passou no exame? " read resposta case "\$resposta" in "sim") echo "Parabéns :)" ;; "não") echo "Não estudou :(" ;; "s") echo "Parabéns :)" ;; "n") echo "Não estudou :(" ;; *) echo "A resposta \$resposta não é válida!" ;; esac</pre>
while	<pre>while [<condição>] do <comandos> done</pre>	<pre>echo "Introduza um nome: " read nome while ["\$nome" != "luis"]; do echo "Não acertou no nome – tente de novo !" read nome done</pre>
until	<pre>until [<condição>] do <comandos> done</pre>	<pre>echo "Escreva o número:" read var echo "Escreva o limite: " read limite until ["\$var" -gt "\$limite"]; do echo "\$var é menor que \$limite." var=\$((var+1)) done</pre>

5. Shell scripts – resolução do guião

P1.

```
#!/bin/bash

echo "Enter 1 to - List files with a user-defined extension"

echo "Enter 2 to - List directories whose size is greater than a
user-defined dimension"

echo "Enter 3 to - List files whose date falls within a time
interval defined by the user"

echo "Enter 4 to - List logins and user names of the system"

echo "Enter 5 to - List files that contain a user-given
expression"

    read case

case $case in

1)

    echo "Enter desired extension: "

        read extension

    ls *.$extension;;

2)

    echo "Enter desired size: "

        read num

    find -type d -size +$(($num))c;;

3)

    echo "Formato: AAAA-MM-DD"

    echo "Introduza data de:"

        read start

    echo "Introduza data ate:"

        read end

    find . -type f -newermt $start ! -newermt $end;;

4)

    while IFS=: read -r f1 f3; do

        echo "Login: $f3"                "Nome de Utilizador: $f1"

    done </etc/passwd;;
```

5)

```
echo "Enter Expression:"
```

```
read word
```

```
grep -r $word;;
```

```
esac
```

```

Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the system
Enter 5 to - List files that contain a user-given expression
4
Login: x:0:0:root:/root:/bin/bash Nome de Utilizador: root
Login: x:1:1:daemon:/usr/sbin:/usr/sbin/nologin Nome de Utilizador: daemon
Login: x:2:2:bin:/bin:/usr/sbin/nologin Nome de Utilizador: bin
Login: x:3:3:sys:/dev:/usr/sbin/nologin Nome de Utilizador: sys
Login: x:4:65534:sync:/bin:/bin/sync Nome de Utilizador: sync
Login: x:5:60:games:/usr/games:/usr/sbin/nologin Nome de Utilizador: games
Login: x:6:12:man:/var/cache/man:/usr/sbin/nologin Nome de Utilizador: man
Login: x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin Nome de Utilizador: lp
Login: x:8:8:mail:/var/mail:/usr/sbin/nologin Nome de Utilizador: mail
Login: x:9:9:news:/var/spool/news:/usr/sbin/nologin Nome de Utilizador: news
Login: x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin Nome de Utilizador: uucp
Login: x:13:13:proxy:/bin:/usr/sbin/nologin Nome de Utilizador: proxy
Login: x:33:33:www-data:/var/www:/usr/sbin/nologin Nome de Utilizador: www-data
Login: x:34:34:backup:/var/backups:/usr/sbin/nologin Nome de Utilizador: backup
Login: x:38:38:Mailng List Manager:/var/list:/usr/sbin/nologin Nome de Utilizador: list
Login: x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin Nome de Utilizador: irc
Login: x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin Nome de Utilizador: gnats
Login: x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin Nome de Utilizador: nobody
Login: x:100:102:systemd Network Management,,:/run/systemd:/usr/sbin/nologin Nome de Utilizador: systemd-network
Login: x:101:103:systemd Resolver,,:/run/systemd:/usr/sbin/nologin Nome de Utilizador: systemd-resolve
Login: x:102:104:systemd Time Synchronization,,:/run/systemd:/usr/sbin/nologin Nome de Utilizador: systemd-timesync
Login: x:103:106:/nonexistent:/usr/sbin/nologin Nome de Utilizador: messagebus
Login: x:104:110:/home/syslog:/usr/sbin/nologin Nome de Utilizador: syslog
Login: x:105:65534:/nonexistent:/usr/sbin/nologin Nome de Utilizador: apt
Login: x:106:111:TPM software stack,,:/var/lib/tpm:/bin/false Nome de Utilizador: tss
Login: x:107:114:/run/uiddd:/usr/sbin/nologin Nome de Utilizador: uiddd
Login: x:108:115:/nonexistent:/usr/sbin/nologin Nome de Utilizador: tcpdump
Login: x:109:116:Avahi autoip daemon,,:/var/lib/avahi-autoipd:/usr/sbin/nologin Nome de Utilizador: avahi-autoipd
Login: x:110:46:usbmux daemon,,:/var/lib/usbmux:/usr/sbin/nologin Nome de Utilizador: usbmux
Login: x:111:117:RealtimeKit,,:/proc:/usr/sbin/nologin Nome de Utilizador: rtkit
Login: x:112:65534:dnsmasq,,:/var/lib/misc:/usr/sbin/nologin Nome de Utilizador: dnsmasq
Login: x:113:120:user for cups-pk-helper service,,:/home/cups-pk-helper:/usr/sbin/nologin Nome de Utilizador: cups-pk-helper
Login: x:114:29:Speech Dispatcher,,:/run/speech-dispatcher:/bin/false Nome de Utilizador: speech-dispatcher
Login: x:115:121:Avahi mDNS daemon,,:/var/run/avahi-daemon:/usr/sbin/nologin Nome de Utilizador: avahi
Login: x:116:65534:Kernel Oops Tracking Daemon,,:/usr/sbin/nologin Nome de Utilizador: kernoops
Login: x:117:123:/var/lib/saned:/usr/sbin/nologin Nome de Utilizador: saned
Login: x:118:124:NetworkManager OpenVPN,,:/var/lib/openvpn/chroot:/usr/sbin/nologin Nome de Utilizador: nm-openvpn
Login: x:119:7:HPLIP system user,,:/run/hplip:/bin/false Nome de Utilizador: hplip
Login: x:120:125:/nonexistent:/bin/false Nome de Utilizador: whoopsie
Login: x:121:126:colord colour management daemon,,:/var/lib/colord:/usr/sbin/nologin Nome de Utilizador: colord
Login: x:122:127:/var/lib/geoclue:/usr/sbin/nologin Nome de Utilizador: geoclue
Login: x:123:128:PulseAudio daemon,,:/var/run/pulse:/usr/sbin/nologin Nome de Utilizador: pulse
Login: x:124:65534:/run/gnome-initial-setup:/bin/false Nome de Utilizador: gnome-initial-setup
Login: x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false Nome de Utilizador: gdm
Login: x:1000:1000:HugoPaixao,,:/home/hugopaixao:/bin/bash Nome de Utilizador: hugopaixao
Login: x:999:999:systemd Core Dumper:/usr/sbin/nologin Nome de Utilizador: systemd-coredump
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$

```

```

hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
total 32
drwxrwxr-x 5 hugopaixao hugopaixao 4096 mar 1 11:11
-rw-rw-r-- 1 hugopaixao hugopaixao 1103 mar 1 11:11
-rw-rw-r-- 1 hugopaixao hugopaixao 39 mar 1 11:11
-rwxrwxr-x 1 hugopaixao hugopaixao 895 mar 1 11:11
drwxrwxr-x 4 hugopaixao hugopaixao 4096 mar 1 11:11
-rw-rw-r-- 1 hugopaixao hugopaixao 85 mar 1 11:11
-rw-rw-r-- 1 hugopaixao hugopaixao 3704 mar 1 11:11
drwxrwxr-x 2 hugopaixao hugopaixao 4096 mar 1 11:11
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
Enter 1 to - List files with a user-defined extension
Enter 2 to - List directories whose size is greater than a user-defined dimension
Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the system
Enter 5 to - List files that contain a user-given expression
2
Enter desired size:
500
.
./Pratica
./Pratica/LabsTest
./Pratica/LabsTest/Lab2
./Pratica/LabsTest/Lab2/C
./Pratica/LabsTest/Lab2/Relatorio
./Pratica/LabsTest/Lab2/Guiao
./Pratica/LabsTest/Lab1
./Pratica/LabsEntrega
./Pratica/LabsEntrega/Lab1
./Pratica/LabsEntrega/Lab1/Scripts
./Pratica/LabsEntrega/Lab1/Scripts/Galeria
./Pratica/LabsEntrega/Lab1/Relatorio
./Pratica/LabsEntrega/Lab1/Guiao
./Aulas
./Aulas/Aula_1
./Aulas/Semana3
./Aulas/Aula_2
./Teoria
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
Enter 1 to - List files with a user-defined extension
Enter 2 to - List directories whose size is greater than a user-defined dimension
Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the system
Enter 5 to - List files that contain a user-given expression
2
Enter desired size:
5000
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$

```

```

hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
Enter 1 to - List files with a user-defined extension
Enter 2 to - List directories whose size is greater than a user-defined dimension
Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the system
Enter 5 to - List files that contain a user-given expression
1
Enter desired extension:
sh
P1_directorySize.sh P1_fileDate.sh P1_login.sh P2.sh P4.sh P6.sh
P1_extension.sh P1_findExpression.sh P1.sh P3.sh P5.sh
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
Enter 1 to - List files with a user-defined extension
Enter 2 to - List directories whose size is greater than a user-defined dimension
Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the system
Enter 5 to - List files that contain a user-given expression
3
Formato: AAAA-MM-DD
Introduza data de:
2021-03-20
Introduza data ate:
2021-03-30
./P1_directorySize.sh
./P1_fileDate.sh
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$

```

Figura 1- Teste 4

```

hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$ ./P1.sh
Enter 1 to - List files with a user-defined extension
Enter 2 to - List directories whose size is greater than a user-defined dimension
Enter 3 to - List files whose date falls within a time interval defined by the user
Enter 4 to - List logins and user names of the sistem
Enter 5 to - List files that contain a user-given expression
5
Enter Expression:
echo
Galeria/P7.sh:echo "Processing image $i ..."
P2.sh:echo "Enter number: "
P2.sh:echo $fatorial
P6.sh:#echo "Introduza o Ano"
P6.sh: echo "Neste ano Fevereiro tem 29 dias"
P6.sh: echo "Neste ano Fevereiro tem 28 dias"
P5.sh: echo "$1 x $begin = $result"
P3.sh:echo "Extensão atual"
P3.sh:echo "Extensão desejada"
P1.sh:echo "Enter 1 to - List files with a user-defined extension"
P1.sh:echo "Enter 2 to - List directories whose size is greater than a user-defined dimension"
P1.sh:echo "Enter 3 to - List files whose date falls within a time interval defined by the user"
P1.sh:echo "Enter 4 to - List logins and user names of the sistem"
P1.sh:echo "Enter 5 to - List files that contain a user-given expression"
P1.sh: echo "Enter desired extension: "
P1.sh: echo "Enter desired size: "
P1.sh: echo "Formato: AAAA-MM-DD"
P1.sh: echo "Introduza data de:"
P1.sh: echo "Introduza data ate:"
P1.sh: echo "Login: $f3" "Nome de Utilizador: $f1"
P1.sh: echo "Enter Expression:"
P1_fileDate.sh:echo "Formato: AAAA-MM-DD"
P1_fileDate.sh:echo "Introduza data de:"
P1_fileDate.sh:echo "Introduza data ate:"
P1_findExpression.sh:echo "Enter Expression:"
P1_login.sh: echo "Login: $f3" "Nome de Utilizador: $f1"
P1_extension.sh:#echo "Extension"
hugopaixao@hugopaixao-VirtualBox:~/Desktop/S0/Pratica/LabsEntrega/Lab1/Scripts$
  
```

P2.

```

#!/bin/bash

echo "Enter number: "

read num

fatorial=1

while [ $num -gt 1 ]
do

fatorial=$((fatorial * num))

num=$((num - 1))
  
```

```

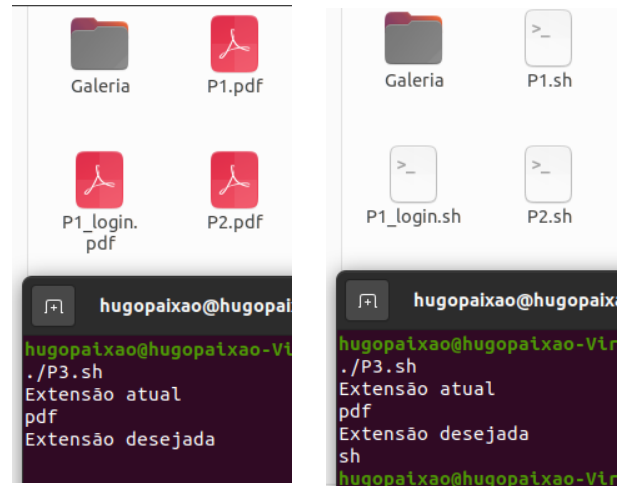
hugopaixao@hugo
./P2.sh
Enter number:
2
2
hugopaixao@hugo
./P2.sh
Enter number:
3
6
hugopaixao@hugo
  
```

done

echo \$fatorial

P3.

```
#!/bin/bash
echo "Extensão atual"
  read now
echo "Extensão desejada"
  read want
for f in *.$now; do
  mv -- "$f" "${f%.$now}.$want"
done
```



P4.

```
#!/bin/bash
echo "Enter files"
    read file1
    read file2
cat $file1 $file2 > file3
wc -l file3
```

P5.

```
#!/bin/bash
begin=0
while [ $begin -le $2 ]
do
    result=$(( $1 * begin ))
    echo "$1 x $begin = $result"
    begin=$(( begin+1 ))
done
```

P6.

```
#!/bin/bash
echo "Introduza o Ano"
    read year
check1=$(( $year % 4 ))
check2=$(( $year % 100 ))
check3=$(( $year % 400 ))
if [ $check1 -eq 0 ] || [ $check2 -eq 0 ] || [ $check3 -eq 0 ];
then
    echo "Neste ano Fevereiro tem 29 dias"
else
    echo "Neste ano Fevereiro tem 28 dias"
fi
```

```
hugopaixao@hugopaixao:~/Desktop/SO/Pratica/LabsEntr$ ./P4.sh
Enter files
P2.sh
P3.sh
24 file3
hugopaixao@hugopaixao:~/Desktop/SO/Pratica/LabsEntr$ wc -l file3
24 file3
```

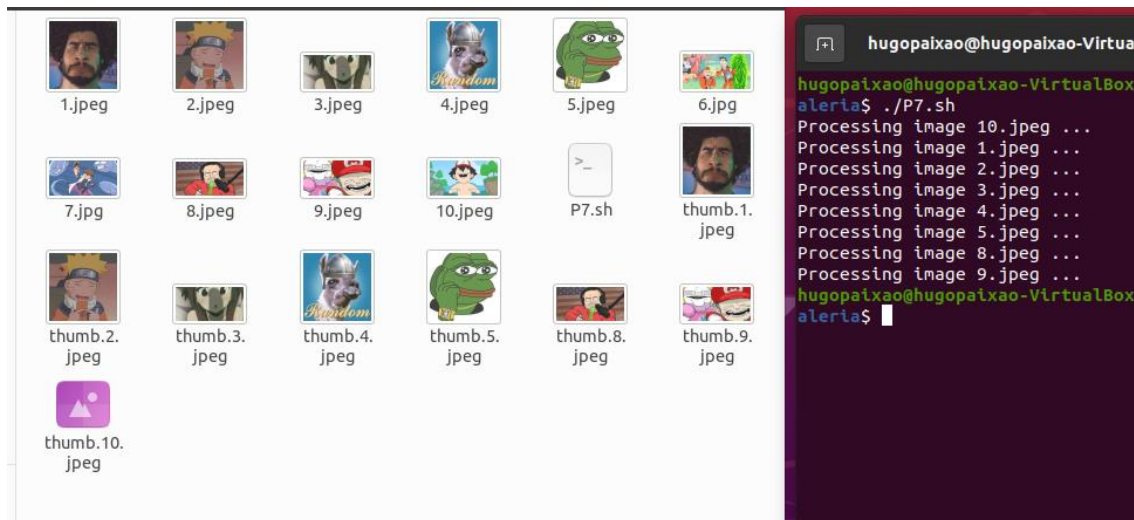
```
hugopaixao@hugopaixao:~/Desktop/SO/Pratica/LabsEntr$ ./P5.sh 5 4
5 x 0 = 0
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
hugopaixao@hugopaixao:~/Desktop/SO/Pratica/LabsEntr$
```

```
hugopaixao@hugopaixao-VirtualBox:~/Desktop/SO/Pratica/LabsEntr$ ./P6.sh
Introduza o Ano
2020
Neste ano Fevereiro tem 29 dias
hugopaixao@hugopaixao-VirtualBox:~/Desktop/SO/Pratica/LabsEntr$ ./P6.sh
Introduza o Ano
2021
Neste ano Fevereiro tem 28 dias
hugopaixao@hugopaixao-VirtualBox:~/Desktop/SO/Pratica/LabsEntr$
```

P7.

```
#!/bin/bash

for i in *.jpeg
do
echo "Processing image $i ..."
convert -thumbnail 200 $i thumb.$i
done
```



6. Conclusão

Com este trabalho foi nos possível desenvolver um entendimento de shell, bash e scripts, apesar de não ser uma linguagem intuitiva mostrou se bastante robusta, e ajudou-nos a expandir um conhecimento sobre a o sistema operativo Linux que já existente era bastante limitado.

7. Webgrafia

<https://web.fe.up.pt/~ssn/2006/peus/peus2006-windows-introducao.pdf>

[https://pt.wikipedia.org/wiki/Shell_\(computação\)](https://pt.wikipedia.org/wiki/Shell_(computação))

<https://searchdatacenter.techtarget.com/definition/shell-script>

<https://pt.wikipedia.org/wiki/Bash>

<https://www.log2base2.com/shell-script-examples/loop/shell-script-to-find-factorial-of-a-number.html>

<https://unix.stackexchange.com/questions/19654/how-do-i-change-the-extension-of-multiple-files>

<https://www.cyberciti.biz/tips/howto-linux-creating-a-image-thumbnails-from-shell-prompt.html>