

Trabalho Prático – Sistemas Operativos

Henrique Sousa Coutinho, Nº 16984

Introdução:

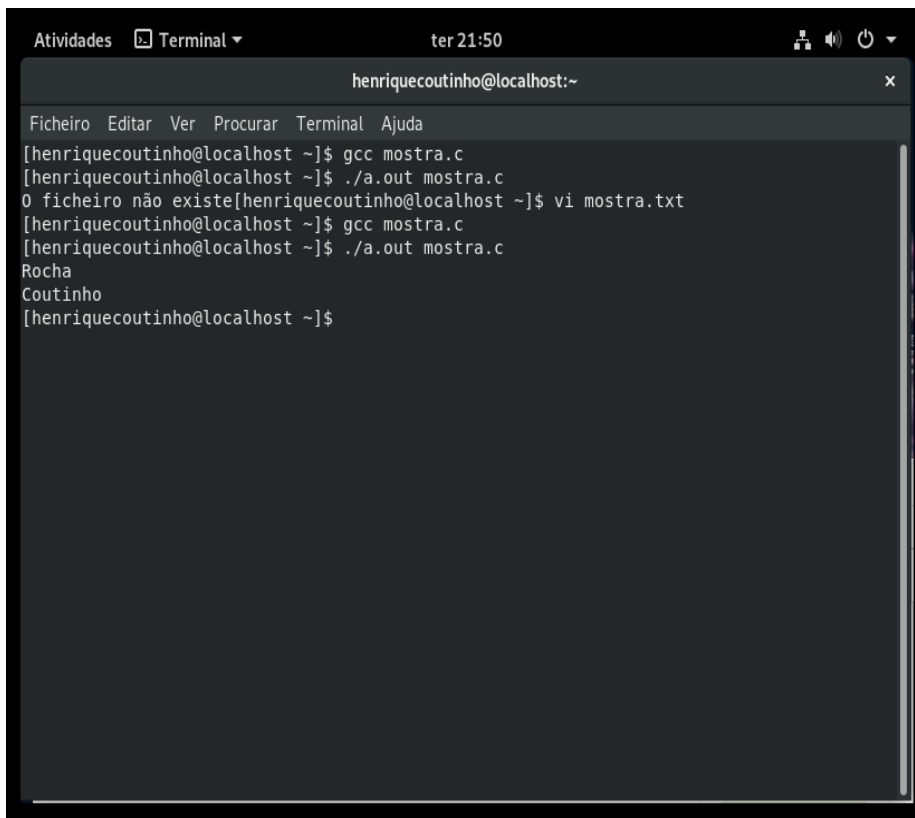
Neste trabalho fui posto à prova quanto aos meus conhecimentos de gestão de processos e de ficheiros, assim como a aplicação da comunicação entre processos aprendidos ao longo das aulas da cadeira de Sistemas Operativos.

Desenvolvimento:

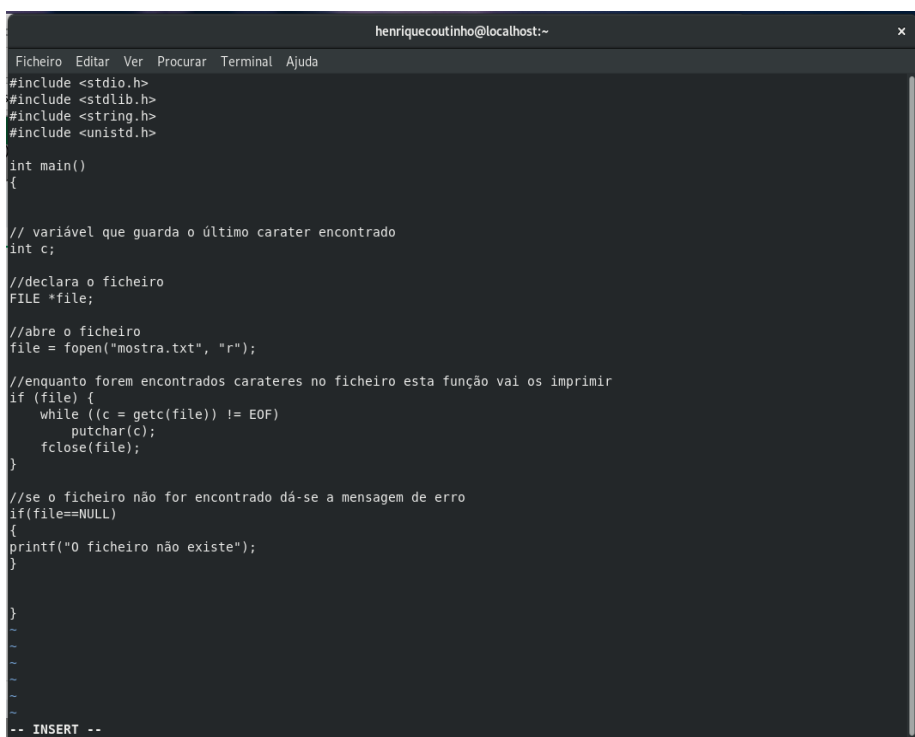
Parte 1

a)

Este comando deve apresentar no ecrã o conteúdo do ficheiro indicado como parâmetro. Caso o ficheiro não exista (na diretoria de trabalho atual), o interpretador deve avisar o utilizador que o ficheiro não existe.



```
Atividades Terminal ter 21:50
henriquecoutho@localhost:~
Ficheiro Editar Ver Procurar Terminal Ajuda
[henriquecoutho@localhost ~]$ gcc mostra.c
[henriquecoutho@localhost ~]$ ./a.out mostra.c
0 ficheiro não existe[henriquecoutho@localhost ~]$ vi mostra.txt
[henriquecoutho@localhost ~]$ gcc mostra.c
[henriquecoutho@localhost ~]$ ./a.out mostra.c
Rocha
Coutho
[henriquecoutho@localhost ~]$
```



```
henriquecoutho@localhost:~
Ficheiro Editar Ver Procurar Terminal Ajuda
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>

int main()
{
    // variável que guarda o último carater encontrado
    int c;

    //declara o ficheiro
    FILE *file;

    //abre o ficheiro
    file = fopen("mostra.txt", "r");

    //enquanto forem encontrados caracteres no ficheiro esta função vai os imprimir
    if (file) {
        while ((c = getc(file)) != EOF)
            putchar(c);
        fclose(file);
    }

    //se o ficheiro não for encontrado dá-se a mensagem de erro
    if(file==NULL)
    {
        printf("0 ficheiro não existe");
    }

}

-- INSERT --
```

b)

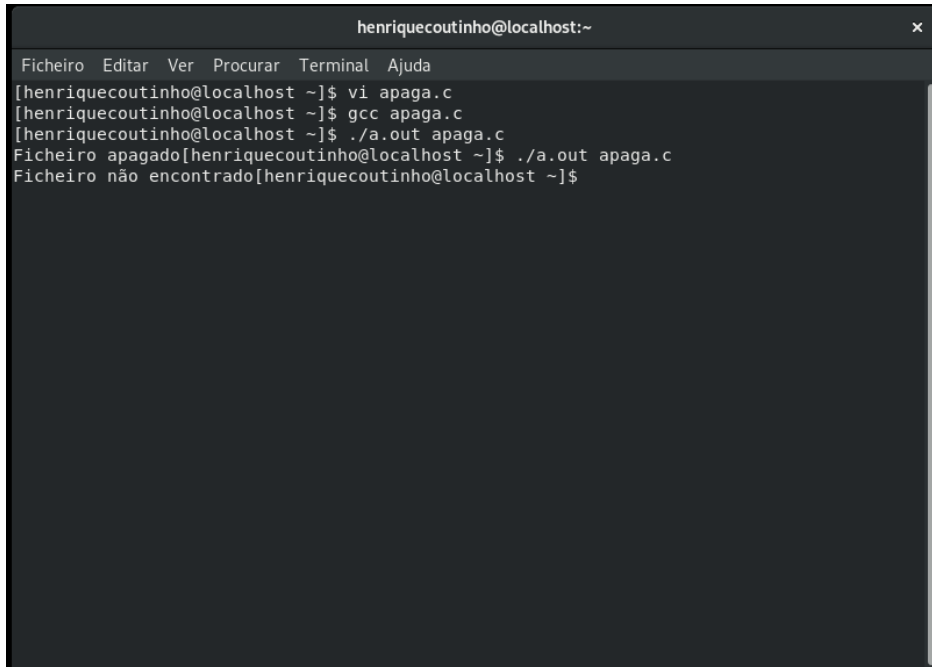
Este comando deve contar o número de linhas de um ficheiro. O número de linhas é definido em Unix como sendo o número de caracteres '\n' existentes no conteúdo do ficheiro. Se o ficheiro não existir, deverá ser indicado ao utilizador uma mensagem de erro.

```
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ vi conta.c  
[henriquecouthinho@localhost ~]$ gcc conta.c  
[henriquecouthinho@localhost ~]$ ./a.out conta.c  
0 ficheiro não existe[henriquecouthinho@localhost ~]$ vi conta.c  
[henriquecouthinho@localhost ~]$ gcc conta.c  
[henriquecouthinho@localhost ~]$ ./a.out conta.c  
0 ficheiro tem 2 linhas [henriquecouthinho@localhost ~]$
```

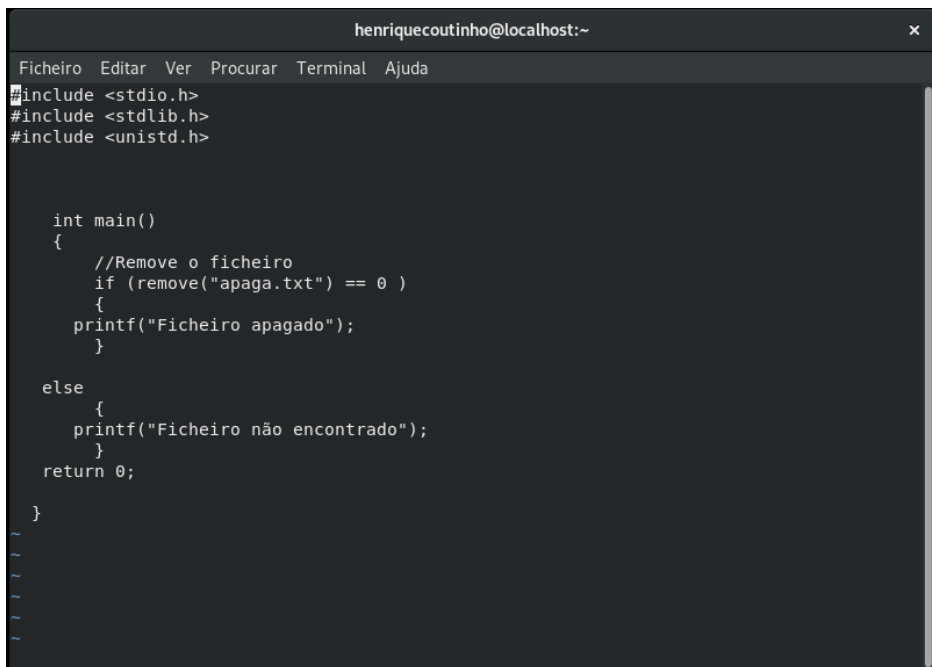
```
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
#include <stdio.h>  
#include <stdlib.h>  
  
int main()  
{  
    FILE *file;  
  
    //variavel para contar as linhas  
    int count = 0;  
  
    //variavel que guarda o ultimo carater encontrado  
    char c;  
  
    // Abre o ficheiro  
    file = fopen("mostra.txt", "r");  
    // Verifica se o ficheiro existe  
    if (file == NULL)  
    {  
        printf("0 ficheiro não existe");  
        return 0;  
    }  
  
    //Apanha os caracteres dentro do ficheiro e guarda os na variavel c  
    for (c = getc(file); c != EOF; c = getc(file))  
    {  
        //incrementa o numero de linhas  
        if (c == '\n')  
            count = count + 1;  
    }  
  
    // Fechar o ficheiro  
    fclose(file);  
  
    //Imprime o numero de linhas  
    printf("0 ficheiro tem %d linhas ", count);  
  
    return 0;  
}  
"conta.c" 41L, 785C
```

c)

Este comando deve apagar o ficheiro com o nome indicado (da diretoria atual). No caso de o ficheiro indicado não existir, e apenas, deve ser apresentado um aviso ao utilizador.



```
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ vi apaga.c  
[henriquecouthinho@localhost ~]$ gcc apaga.c  
[henriquecouthinho@localhost ~]$ ./a.out apaga.c  
Ficheiro apagado[henriquecouthinho@localhost ~]$ ./a.out apaga.c  
Ficheiro não encontrado[henriquecouthinho@localhost ~]$
```



```
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
  
int main()  
{  
    //Remove o ficheiro  
    if (remove("apaga.txt") == 0 )  
    {  
        printf("Ficheiro apagado");  
    }  
  
    else  
    {  
        printf("Ficheiro não encontrado");  
    }  
    return 0;  
}
```

d)

Este comando apresenta a informação do sistema de ficheiros em relação ao ficheiro dado, como o seu tamanho, tipo de ficheiro (normal, diretoria, link, etc.), i-node, utilizador dono, data de criação e da última modificação.

```
Ficheiro Editar Ver Procurar Terminal Ajuda
[henriquecouthinho@localhost ~]$ vi informa.c
[henriquecouthinho@localhost ~]$ gcc informa.c
[henriquecouthinho@localhost ~]$ ./a.out informa.c
0 ficheiro não foi encontrado[henriquecouthinho@localhost ~]$ vi informa.c
[henriquecouthinho@localhost ~]$ gcc informa.c
[henriquecouthinho@localhost ~]$ ./a.out informa.c
Tamanho do Ficheiro:
 15 bytes
Inode do ficheiro:
 821244
Ultimo Acesso:
 Wed Jun 19 22:14:45 2019
Ultima Modificação:
 Wed Jun 19 22:14:45 2019
Dono do ficheiro:
 henriquecouthinho
Grupo do ficheiro:
 henriquecouthinho
Tipo de ficheiro:
 ficheiro regular
[henriquecouthinho@localhost ~]$
```

```
Ficheiro Editar Ver Procurar Terminal Ajuda
#include <stdio.h>
#include <sys/stat.h>
#include <time.h>
#include <pwd.h>
#include <grp.h>
#include <stdlib.h>

//comandos para imprimir as funções descritas embaixo
#define comando "stat -c %U mostra.txt"
#define comandog "stat -c %G mostra.txt"
#define comandof "stat -c %F mostra.txt"

//estrutura de atributos predefinida
struct stat info;

int main()
{
    //verifica se o ficheiro existe
    if(stat("mostra.txt", &info) == 0)
    {
        printf("Tamanho do Ficheiro:\n %i bytes \n", info.st_size);
        printf("Inode do ficheiro:\n %i \n", info.st_ino);
        printf("Ultimo Acesso:\n %s", ctime(&info.st_atime));
        printf("Ultima Modificação:\n %sDono do ficheiro:\n", ctime(&info.st_mtime));
        //chama o comando para mostrar o dono do ficheiro
        system(comando);
        printf("Grupo do ficheiro:\n");
        //chama comando para mostrar grupo do ficheiro
        system(comandog);
        //chama o comando para mostrar o tipo do ficheiro
        printf("Tipo de ficheiro:\n");
        system(comandof);
    }

    else

    {

        printf("0 ficheiro não foi encontrado");
    }

}
```

e)

Este comando deve acrescentar o conteúdo do ficheiro2 no final do ficheiro1. Caso algum dos ficheiros não exista, deve ser apresentado um aviso ao utilizador.

```
henriquecoutho@localhost:~  
Ficheiro  Editar  Ver  Procurar  Terminal  Ajuda  
[henriquecoutho@localhost ~]$ cat ficheiro1.txt  
Rocha  
[henriquecoutho@localhost ~]$ cat ficheiro2.txt  
Coutho  
[henriquecoutho@localhost ~]$ vi acrescenta.c  
[henriquecoutho@localhost ~]$ gcc acrescenta.c  
[henriquecoutho@localhost ~]$ ./a.out acrescenta.c  
Os ficheiros não foram encontrados[henriquecoutho@localhost ~]$ vi acrescenta.c  
[henriquecoutho@localhost ~]$ gcc acrescenta.c  
[henriquecoutho@localhost ~]$ ./a.out acrescenta.c  
O ficheiro 2 foi acrescentado ao ficheiro1[henriquecoutho@localhost ~]$ cat ficheiro1.txt  
Rocha  
Coutho  
[henriquecoutho@localhost ~]$ cat ficheiro2.txt  
Coutho  
[henriquecoutho@localhost ~]$
```

```
henriquecoutho@localhost:~  
Ficheiro  Editar  Ver  Procurar  Terminal  Ajuda  
#include <stdlib.h>  
  
int main()  
{  
    FILE *f1;  
    FILE *f2;  
    char c;  
  
    //Os ficheiros são abertos  
    f1=fopen("ficheiro1.txt", "r");  
    f2=fopen("ficheiro2.txt", "r");  
  
    //Condição ficheiros não existentes  
    if(f1 == NULL || f2 == NULL)  
    {  
        printf("Os ficheiros não foram encontrados");  
        //O programa é fechado  
        exit(2);  
    }  
    else  
    {  
  
        //O ficheiro fica em modo append para permitir a escrita  
        f1=fopen("ficheiro1.txt", "a");  
  
        //Enquanto houverem caracteres e o ficheiro não chegar ao fim os seus caracteres são guardados no c  
        while ((c = fgetc(f2)) != EOF)  
        //conteudo de c vai para o f1  
        fputc(c, f1);  
        printf("O ficheiro 2 foi acrescentado ao ficheiro1");  
    }  
  
    //os ficheiros são fechados  
    fclose(f1);  
    fclose(f2);  
  
    return 0;  
}
```

Parte 2

a)

Utilizando o comando cat na Bash junte todos os ficheiros num único ficheiro chamado “todotexto.txt”.

O comando utilizado foi o “cat”.

```
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ ls  
acrescenta.c      ficheiro2.txt      ipca-sort3.txt     sosd.c  
a.out            fork               ipca-sorted.txt    t1.txt  
apaga.c          fork.c            ipca-uniq1.txt     t2.txt  
apaga.txt        getpid.c          ipca-uniq2.txt     t3.txt  
'Área de Trabalho'  Imagens           Modelos            t4.txt  
conta.c          informa.c         mostra.c           Transferências  
Documentos       ipca-cat.txt      mostra.txt         Videos  
exec.c           ipca-resultado2.txt  Música            wait.c  
execl.c          ipca-sort1-ordenado.txt  nohup.out         while.c  
execve.c         ipca-sort1.txt     Público  
ficheiro1.txt    ipca-sort2.txt     sosd  
[henriquecouthinho@localhost ~]$ cat t1.txt t2.txt t3.txt t4.txt > todotexto.txt  
[henriquecouthinho@localhost ~]$
```

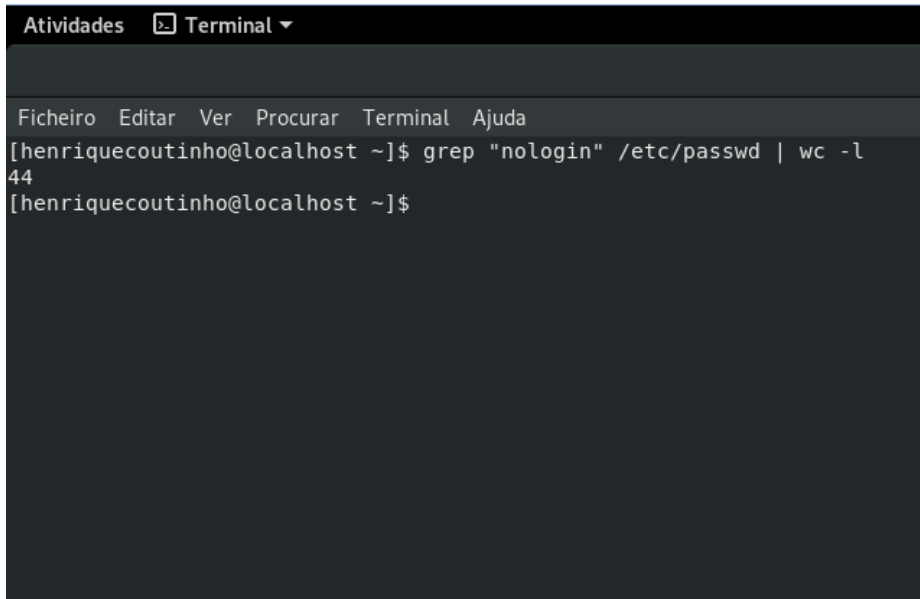
b)

Utilize o comando grep para filtrar as linhas do ficheiro /etc/passwd que contêm o texto “nologin”.

```
Atividades Terminal sex 16:24  
henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ grep "nologin" /etc/passwd  
bin:x:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin  
adm:x:3:4:adm:/var/adm:/sbin/nologin  
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin  
operator:x:11:0:operator:/root:/sbin/nologin  
games:x:12:100:games:/usr/games:/sbin/nologin  
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin  
nobody:x:65534:65534:Kernel Overflow User:/sbin/nologin  
dbus:x:81:81:System message bus:/sbin/nologin  
systemd-coredump:x:999:997:systemd Core Dumper:/sbin/nologin  
systemd-network:x:192:192:systemd Network Management:/sbin/nologin  
systemd-resolve:x:193:193:systemd Resolver:/sbin/nologin  
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin  
polkitd:x:998:996:User for polkitd:/sbin/nologin  
gluster:x:997:994:GlusterFS daemons:/run/gluster:/sbin/nologin  
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin  
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin  
qemu:x:107:107:qemu user:/sbin/nologin  
nm-openconnect:x:996:990:NetworkManager user for OpenConnect:/sbin/nologin  
unbound:x:995:989:Unbound DNS resolver:/etc/unbound:/sbin/nologin  
usbmuxd:x:113:113:usbmuxd user:/sbin/nologin  
chrony:x:994:988:/var/lib/chrony:/sbin/nologin  
geoclue:x:993:987:User for geoclue:/var/lib/geoclue:/sbin/nologin  
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin  
pipewire:x:992:986:PipeWire System Daemon:/var/run/pipewire:/sbin/nologin  
saslauthd:x:991:76:Saslauthd user:/run/saslauthd:/sbin/nologin  
dnsmasq:x:985:985:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin  
raddvd:x:75:75:raddvd user:/sbin/nologin  
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin  
openvpn:x:984:982:OpenVPN:/etc/openvpn:/sbin/nologin  
nm-openvpn:x:983:981:Default user for running openvpn spawned by NetworkManager:/sbin/nologin  
abrt:x:173:173:/etc/abrt:/sbin/nologin  
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin  
colord:x:982:980:User for colord:/var/lib/colord:/sbin/nologin  
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin  
gdm:x:42:42:/var/lib/gdm:/sbin/nologin  
gnome-initial-setup:x:981:979:/run/gnome-initial-setup:/sbin/nologin  
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin  
vboxadd:x:980:1:/var/run/vboxadd:/sbin/nologin  
tcpdump:x:72:72:/sbin/nologin  
nginx:x:979:977:Nginx web server:/var/lib/nginx:/sbin/nologin  
squid:x:23:23:/var/spool/squid:/sbin/nologin  
webalizer:x:67:976:Webalizer:/var/www/usage:/sbin/nologin  
[henriquecouthinho@localhost ~]$
```

c)

Utilizando o comando `wc` conte o número de linhas resultado do exercício anterior.



```
Atividades Terminal ▾  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ grep "nologin" /etc/passwd | wc -l  
44  
[henriquecouthinho@localhost ~]$
```

d)

Com os ficheiros descarregados em a), apresente o comando de bash que fazendo uso de pipes (`|`) junta os ficheiros novamente utilizando o comando `cat` e ordena esse resultado com o comando `sort`, navegando no resultado com o comando `less`.



```
Atividades Terminal ▾ sex 16:58 • henriquecouthinho@localhost:~  
Ficheiro Editar Ver Procurar Terminal Ajuda  
[henriquecouthinho@localhost ~]$ cat t1.txt t2.txt t3.txt t4.txt > todotexto.txt | sort > todotexto.txt; cat todotexto.txt | less  
[henriquecouthinho@localhost ~]$
```


a)

Crie uma máquina virtual Linux. Na máquina virtual, crie uma partição adicional de 10GB acessível através do caminho /home. Apresente os comandos utilizados para a criação da partição de 10 GB e a tabela de partições executando o comando “fdisk -l” como administrador (utilizador root) e as partições montadas através do “mount”.

Comandos utilizados:

- fdisk -l

Lista no terminal os discos e partições existentes no sistema

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	2099199	2097152	1G	83	Linux
/dev/sda2		2099200	23070719	20971520	10G	83	Linux

- fdisk /dev/sda

Comando para configurar o disco

- n

Para criar uma nova partição

- p

Partição primária

- 2099200 e +10G

Comandos para delimitar o tamanho da partição

```
Command (m for help): n
Partition type
  p   primary (1 primary, 0 extended, 3 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (2099200-83886079, default 2099200): 2099200
Last sector, +sectors or +size{K,M,G,T,P} (2099200-83886079, default 83886079): +10G

Created a new partition 2 of type 'Linux' and of size 10 GiB.
```