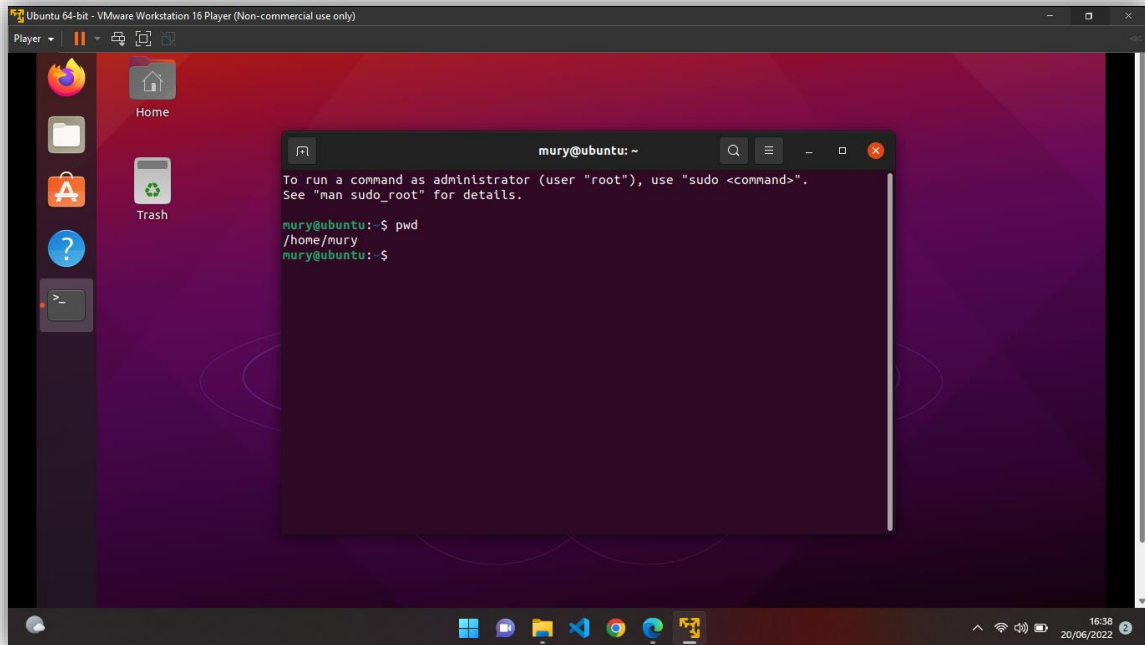


Prática de Laboratório – Comandos Linux – Nível Básico

1. Utilize o comando `pwd` para saber o nome do diretório em que você se encontra.



2. Utilize o comando `ls` para listar o conteúdo do local onde você está.

```
mury@ubuntu:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
mury@ubuntu:~$
```

3. Leia o manual de comandos do `ls` e descubra como listar os arquivos ordenados pela data da última atualização.

```
mury@ubuntu:~$ ls -c -lt
total 32
drwxr-xr-x 2 mury mury 4096 Jun 20 12:42 Documents
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Desktop
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Downloads
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Music
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Pictures
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Public
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Templates
drwxr-xr-x 2 mury mury 4096 Jun 18 08:32 Videos
```

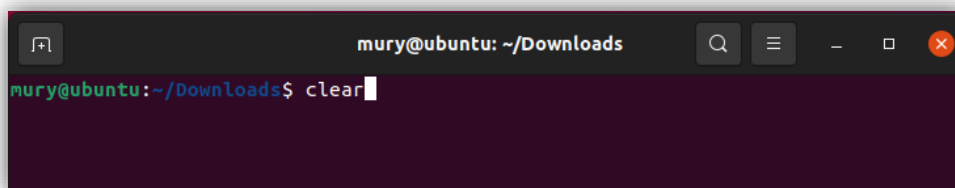
4. Navegue para dentro do diretório Downloads com o comando.

```
mury@ubuntu:~$ cd Downloads
mury@ubuntu:~/Downloads$
```

5. Digite novamente os comandos `pwd` e `ls` para saber onde você está e o que está disponível neste local

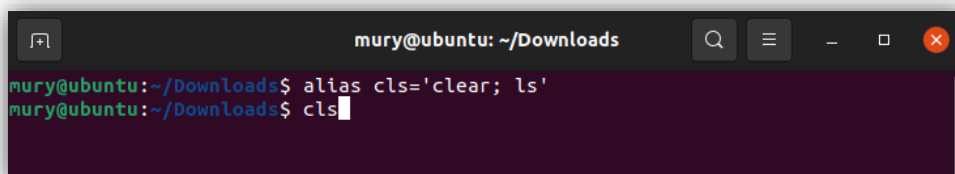
```
mury@ubuntu:~$ cd Downloads
mury@ubuntu:~/Downloads$ pwd
/home/mury/Downloads
mury@ubuntu:~/Downloads$ ls
mury@ubuntu:~/Downloads$
```

6. Use o comando `clear` para limpar a sua janela de comandos.

A terminal window titled 'mury@ubuntu: ~/Downloads' with search, menu, and window control icons. The prompt is 'mury@ubuntu:~/Downloads\$' and the command 'clear' has been entered, resulting in a blank terminal screen.

```
mury@ubuntu:~/Downloads$ clear
```

7. Crie o comando clear screen (vulgo `cls`) com a instrução: `alias cls='clear; ls'`

A terminal window titled 'mury@ubuntu: ~/Downloads' with search, menu, and window control icons. The prompt is 'mury@ubuntu:~/Downloads\$' and the command 'alias cls='clear; ls'' has been entered. The next line shows the prompt 'mury@ubuntu:~/Downloads\$' followed by 'cls' entered.

```
mury@ubuntu:~/Downloads$ alias cls='clear; ls'
mury@ubuntu:~/Downloads$ cls
```

8. Redefina o comando acima para incluir a listagem do nome do diretório no conjunto de comandos anterior.

```
mury@ubuntu:~/Downloads$ alias cls='clear; pwd; ls'
mury@ubuntu:~/Downloads$
```

9. Execute o comando: `ping www.dropbox.com`

```

mury@ubuntu:~/Downloads$ ping www.dropbox.com
PING www-env.dropbox-dns.com (162.125.5.18) 56(84) bytes of data.
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=1 ttl=128 time=143 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=2 ttl=128 time=146 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=3 ttl=128 time=141 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=4 ttl=128 time=146 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=5 ttl=128 time=143 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=6 ttl=128 time=144 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=7 ttl=128 time=144 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=8 ttl=128 time=143 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=9 ttl=128 time=144 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=10 ttl=128 time=144 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=11 ttl=128 time=147 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=12 ttl=128 time=154 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=13 ttl=128 time=147 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=14 ttl=128 time=141 ms
64 bytes from 162.125.5.18 (162.125.5.18): icmp_seq=15 ttl=128 time=146 ms
^C
--- www-env.dropbox-dns.com ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14283ms
rtt min/avg/max/mdev = 141.363/144.809/153.545/2.977 ms
mury@ubuntu:~/Downloads$

```

10. Agora faremos o download de um arquivo oferecido neste servidor. Para tanto vamos utilizar o comando `wget`. O link para download está no comando abaixo, basta copiar, colar e em seguida digitar o Enter para executar.

```

mury@ubuntu: ~/Downloads
mury@ubuntu:~/Downloads$ wget https://www.dropbox.com/s/2kdb1j3b477ygyv/sample.tar.gz
--2022-06-20 13:55:36-- https://www.dropbox.com/s/2kdb1j3b477ygyv/sample.tar.gz
Resolving www.dropbox.com (www.dropbox.com)... 162.125.5.18, 2620:100:601d:18::a27d:512
Connecting to www.dropbox.com (www.dropbox.com)|162.125.5.18|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: /s/raw/2kdb1j3b477ygyv/sample.tar.gz [following]
--2022-06-20 13:55:37-- https://www.dropbox.com/s/raw/2kdb1j3b477ygyv/sample.tar.gz
Reusing existing connection to www.dropbox.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com/cd/0/inline/BnlyLOFIoVIQuXN-jElyu00BqomJ9nUn3aDGUNr5dLj8IH6YNMXEfbSvrZKeuSESEvaPyxlocvKERPMWX9wUVu3tX8aph95chw-ZSdJUKk3pBhxeeuuew-VD2JKIMYeu62afmJHdT-gp5SoTeVavzZ6vCXBPAGZ6AGL5qU9r8DIAiw/file# [following]
--2022-06-20 13:55:37-- https://ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com/cd/0/inline/BnlyLOFIoVIQuXN-jElyu00BqomJ9nUn3aDGUNr5dLj8IH6YNMXEfbSvrZKeuSESEvaPyxlocvKERPMWX9wUVu3tX8aph95chw-ZSdJUKk3pBhxeeuuew-VD2JKIMYeu62afmJHdT-gp5SoTeVavzZ6vCXBPAGZ6AGL5qU9r8DIAiw/file
Resolving ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com (ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com)... 162.125.5.15, 2620:100:601d:15::a27d:50f
Connecting to ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com (ucb7c81cb3049da5eec870d96fa9.dl.dropboxusercontent.com)|162.125.5.15|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: /cd/0/inline2/BnlyLOFIoVIQuXN-jElyu00BqomJ9nUn3aDGUNr5dLj8IH6YNMXEfbSvrZKeuSESEvaPyxlocvKERPMWX9wUVu3tX8aph95chw-ZSdJUKk3pBhxeeuuew-VD2JKIMYeu62afmJHdT-gp5SoTeVavzZ6vCXBPAGZ6AGL5qU9r8DIAiw/file
5vhf43iHtIeQX0EehM6SwsE_ATbDZ6IKDXjWTD_ax0xfoJDRW_HzYearL72Qt51mB6hA9L2G04aC8FW04Hs2RPI-

```

11. O arquivo `sample.tar.gz` está compactado. Para extrair o seu conteúdo teremos que utilizar o comando `tar`. Curiosamente o comando `tar` também é usado para compactar, mas com os parâmetros certos ele fara a descompactação ou compactação. Neste caso digite o comando abaixo para descompactar.

```
mury@ubuntu: ~/Downloads
/home/mury/Downloads
sample.tar.gz
mury@ubuntu:~/Downloads$ tar -xvf sample.tar.gz
sample/
sample/andromeda.jpg
sample/SOAPdenovo2.pdf
sample/sequence.fasta
sample/Poster.pdf
sample/pan-genome-size.pdf
sample/proteinas/
sample/proteinas/Rv0001
sample/proteinas/Rv0002
sample/proteinas/Rv0003
sample/proteinas/Rv0004
sample/proteinas/Rv0005
sample/proteinas/Rv0006
sample/proteinas/Rv0007
sample/proteinas/Rv0009
sample/proteinas/Rv0012
sample/proteinas/Rv0013
sample/proteinas/Rv0023
sample/proteinas/Rv0024
sample/proteinas/Rv0025
sample/proteinas/Rv0026
```

12. Descubra também qual parâmetro faria a compactação de um arquivo.

```
mury@ubuntu: ~/Downloads
mury@ubuntu:~/Downloads$ tar -czvf compactado.tar.gz sample
sample/
sample/andromeda.jpg
sample/SOAPdenovo2.pdf
sample/sequence.fasta
sample/proteinas/
sample/proteinas/Rv0296c
sample/proteinas/Rv1283c
sample/proteinas/Rv0054
sample/proteinas/Rv3143
sample/proteinas/Rv0529
sample/proteinas/Rv3188
sample/proteinas/Rv1882c
sample/proteinas/Rv0408
sample/proteinas/Rv2675c
sample/proteinas/Rv2866
sample/proteinas/Rv3237c
sample/proteinas/Rv1206
sample/proteinas/Rv3122
sample/proteinas/Rv1430
sample/proteinas/Rv2737A
sample/proteinas/Rv2337c
sample/proteinas/Rv0868c
sample/proteinas/Rv0010c
```

13. Agora temos um diretório novo chamado sample dentro da pasta Downloads. Posicione o nosso terminal de comandos dentro deste diretório/pasta com o comando `cd sample`.

```
mury@ubuntu:~/Downloads$ cd sample/
mury@ubuntu:~/Downloads/sample$
```

14. Execute o nosso comando `cls` dentro da pasta sample.

```
/home/mury/Downloads/sample
andromeda.jpg      Poster.pdf      sequence.fasta
pan-genome-size.pdf  proteinas      SOAPdenovo2.pdf
mury@ubuntu:~/Downloads/sample$
```

15. Veja que temos vários arquivos. Alguns são binários e outros são textos. Uma maneira de exibir o conteúdo de arquivos texto de forma rápida é usando o comando `cat`. Execute o comando `cat sequence.fasta`

```
mury@ubuntu:~/Downloads/sample$ cat sequence.fasta
>Cp1002_0126a
MHFKTRMSLFCATTATTAATSLAVASLQPAAAVEQPSNTIVSTIMLPTKATVTKTFTVSSTKGTARADYSSN
SITVQPGDQITSVKIHSQGGYTEFSELTEFVPSVGRLLHESITTFKEGDSGPHPLKVAGWNATSQADRVTFR
TNDGPKAKAITLDTTLEYTYVGVRATGDPSTRFQLSSSDSNTVFTSASGPKIHVKKLTLPWLWLGAFPGAIF
FDSLTLNLLSPILRALNIL
>Cp1002_1802
MLFPSRFQGTFLKPLITAALAVFCVGFPTATAQVIPYTDPDGFYTSIPSAENTTPGTVLSQRDVPMPVLD
VLVKKMIRIAYTSTHPNGFSTPVTGAVLLPTAPWRGPGRPVALLAPGTQAGAGDSCAPSKLLTMGGEYEMF
SAAALLNAGWTVAVTDYQGLGTPGNHTYMNRKAQGAALLDLGRAITTLNLPDVNNHTPIIPWGSQGGGA
SAAAAEMHRAYAPDVNVVLAYAGGV PANLLSVSSSLEGTALTGALGYVITGMYEIYPEIREPIHNF LNTR
GQVWLDQTSRDCLPESLLTMPLPDTSLTVSQRLTSLSDDVFORAISEQOIGLTAPDIPVFVAQGLND
GIIPAEQARIMVNGWLSQGADVITYWEDPSPALDKLSGHIHVLASSFLPAVEWAEQRLAALGQPTP
```

16. Utilize o mesmo comando para listar o arquivo Poster.pdf e observe o que será exibido.

```
mury@ubuntu: ~/Downloads/sample
~#000W0~|G!-lpx08bee-0eRE0000"0遇<:P00000+0#zw#000P00000;0a0S000{0il[G0-000.00aXN
0;00050-6W00BieF7ec0a.0<0B90XX=00000A<K00000Z00HYGe0000:00
W002ay0000za0m0ueA0!000000
k00I00E00hZ0FDR|-x_0G7"6%00000Hj+00I09000#007nCV02HR$000&0f=0m[bW#00kk0k00000<0jL0;^0000W000
00t7Tut0+ev000$00@|m001-B0绍000000s0/80S600000, G600000.op\00-0aUT0:t0/yV000000c00
0Ts'000O1(0feLX 0000]0<c0S000aJ00000000000000000Y00b0000000K0S\0z+x0f0^
010000C0&0 $00C00q100 :AdDBFb5?00m0000 00CI,=000001VB6h>0400hbZM0&00tD0(%010000TZ00.uJ'Ih0B0
Y0
E?vLL0{00000=I00z0@00VC0 %v0000
D
0000U000R00Tj
0*AL
0000X0<0*0i0D$0_00spzp0\000G0050000$Sq00\0i0000=0{
w0000UT0070q3}0fk0;"0%N0)0O30+0;"0"00b0 :02;g0^00yQ4%JJcD+10Ni0%1070Yf
[%5]T#00fll
Bn"/a<B>00a0000Ar0a$*<$<#0M>&_0300EMr000ek0g 000,0KP01z0dc0n<m0J0000q0.0a0000 00000y0
B0000(T)0G0=000000G000PL000
0p0000000000H000.00edzT Z*0t00'040:k00
000:Epo0$0璦00Wp5<00W
0W0L0
/00 ?<000F050pM00:|0>00-0-R0-0-Y0V000000etno`
[!
W \va0
0J00}@0 A0
]000[0W0000
000000900Ket^0$03-0}00c0-0-Z0000
```

17. Adianto que dentro da pasta proteínas (sem acento) existem vários arquivos. Vamos listar estes arquivos com o comando ls. Porém não peço que você se posicione dentro da pasta proteínas, estando dentro da pasta sample podemos fazer isso. Para tanto digite o comando `ls proteínas`

```
mury@ubuntu: ~/Downloads/sample
mury@ubuntu:~/Downloads/sample$ ls proteinas/
Rv0001  Rv0397  Rv0784  Rv1177  Rv1569  Rv1965  Rv2356c  Rv2749  Rv3144c  Rv3534c
Rv0002  Rv0398c  Rv0785  Rv1178  Rv1570  Rv1966  Rv2357c  Rv2750  Rv3145  Rv3535c
Rv0003  Rv0399c  Rv0786c  Rv1179c  Rv1571  Rv1967  Rv2358  Rv2751  Rv3146  Rv3536c
Rv0004  Rv0400c  Rv0787  Rv1180  Rv1572c  Rv1968  Rv2359  Rv2752c  Rv3147  Rv3537
Rv0005  Rv0401  Rv0787A  Rv1181  Rv1573  Rv1969  Rv2360c  Rv2753c  Rv3148  Rv3538
Rv0006  Rv0402c  Rv0788  Rv1182  Rv1574  Rv1970  Rv2361c  Rv2754c  Rv3149  Rv3539
Rv0007  Rv0403c  Rv0789c  Rv1183  Rv1575  Rv1971  Rv2362c  Rv2755c  Rv3150  Rv3540c
Rv0008c  Rv0404  Rv0790c  Rv1184c  Rv1576c  Rv1972  Rv2363  Rv2756c  Rv3151  Rv3541c
Rv0009  Rv0405  Rv0791c  Rv1185c  Rv1577c  Rv1973  Rv2364c  Rv2757c  Rv3152  Rv3542c
Rv0010c  Rv0406c  Rv0792c  Rv1186c  Rv1578c  Rv1974  Rv2365c  Rv2758c  Rv3153  Rv3543c
Rv0011c  Rv0407  Rv0793  Rv1187  Rv1579c  Rv1975  Rv2366c  Rv2759c  Rv3154  Rv3544c
Rv0012  Rv0408  Rv0794c  Rv1188  Rv1580c  Rv1976c  Rv2367c  Rv2760c  Rv3155  Rv3545c
Rv0013  Rv0409  Rv0795  Rv1189  Rv1581c  Rv1977  Rv2368c  Rv2761c  Rv3156  Rv3546
Rv0014c  Rv0410c  Rv0796  Rv1190  Rv1582c  Rv1978  Rv2369c  Rv2762c  Rv3157  Rv3547
Rv0015c  Rv0411c  Rv0797  Rv1191  Rv1583c  Rv1979c  Rv2370c  Rv2763c  Rv3158  Rv3548c
Rv0016c  Rv0412c  Rv0798c  Rv1192  Rv1584c  Rv1980c  Rv2371  Rv2764c  Rv3159c  Rv3549c
Rv0017c  Rv0413  Rv0799c  Rv1193  Rv1585c  Rv1981c  Rv2372c  Rv2765  Rv3160c  Rv3550
Rv0018c  Rv0414c  Rv0800  Rv1194c  Rv1586c  Rv1982c  Rv2373c  Rv2766c  Rv3161c  Rv3551
Rv0019c  Rv0415  Rv0801  Rv1195  Rv1587c  Rv1983  Rv2374c  Rv2767c  Rv3162c  Rv3552
Rv0020c  Rv0416  Rv0802c  Rv1196  Rv1588c  Rv1984c  Rv2375  Rv2768c  Rv3163c  Rv3553
Rv0021c  Rv0417  Rv0803  Rv1197  Rv1589  Rv1985c  Rv2376c  Rv2769c  Rv3164c  Rv3554
Rv0022c  Rv0418  Rv0804  Rv1198  Rv1590  Rv1986  Rv2377c  Rv2770c  Rv3165c  Rv3555c
Rv0023  Rv0419  Rv0805  Rv1199c  Rv1591  Rv1987  Rv2378c  Rv2771c  Rv3166c  Rv3556c
```

18. Digite o comando `cat proteinas/* > mt.fasta`. Esse comando lista o conteúdo de todos os arquivos dentro da pasta `proteinas` (`proteinas/*`) e redireciona o conteúdo desta listagem (sinal de maior `>`) para dentro de um novo arquivo cujo nome determinamos que seria `mt.fasta`.

```
mury@ubuntu:~/Downloads/sample$ cat proteinas/* > mt.fasta
mury@ubuntu:~/Downloads/sample$
```

19. Agora tente listar o conteúdo do arquivo `mt.fasta` utilizando o comando `cat`.

```
mury@ubuntu: ~/Downloads/sample
VERFVEIGVKSSPTVAGLATNTLKLPEYAHSTVEVLNAERDAAVLFATDTPPEPEPEEDE
PVAESPAPDVVSEAPVAPAASSAGPRPDDLVDAAADATLALIALSAKMRIDQIEELDSI
ESITDGASSRRNQLLVDLGSLENLGAIDGAAESDLAGLRSQVTKLARTYKPYGPLVSDAI
NDQLRTVLGPSGRKPGAIAERVKKTWELGEGWAKHVTVEVALGTREGSSVRGGAMGHLHE
GALADAASVDKVIDAAVASVAARQGVSVLPSAGSGGGATIDAAALSEFTDQITGREGVL
ASARLVLQQLGLDDPVNALPAAPDSELIDLVTAE LGADWPRLVAPVDFPKKAVVDDRHW
ASAREDLVKLWLTDEGDIDADWPRLAERFEGAGHVATQATWQKSLAAGRQIHASLYG
RIAAGAENPEPGRYGGEVAVVTGASKGSIAASVVARLLDGGATVIATTSKLDEERLAFYR
TLYRDHARYGAALWLVAANMASYSDDVALVEHIGTEQTESLGPQSIHIKDAQPTLLFPF
AAPRVVGDLSAAGSRAEMEMKVLWAVQRLIGGLSTIGAERDIASRLHVVLPGSPNRGMF
GGDGAYGEAKSALDAVSVRWHAESSWAARVSLAHALIGWTRGTGLMGHNDIVAAVEEAG
VTYSTDEMAALLLDLCAESKVAAARSPKADLTGGLAEANLDMAEAAKAREQMSAAA
AVDEDEAPGAIAALPSPPRGFTPAPPPQWDDLVDPADLVIVGGAEIGPYGSSRTRFE
MEVENELSAAGVLELAWTTGLIRWEDDPQPGHYDTESEGMVDESELVQRYHDAVQVRVGI
REFVDDGAIDPDHASPLLVSFLEKDFAFVVSSEADARAFVEFDPEHTVIRVPDSTDHQQ
VIRKAGTEIRVPRKTLKSRVVGQIPTGFDPVTWGISADMAGSIDRLAVWMMVATVDAFL
SSGFSAPAEVMRYVHPSLVANTQGTGMGGTSMQTHYHGNLLGRNKPNDIFQEVLPIIAA
HVVQSYVSGYAMIHVPAACATAAVSVEEGVDKIRLGKAQLVAGGLDDLLEGIIIGFD
MAATADTSMCMCGRIHDSKFSRPNDRRLGFVEAQGGGTILLARGDLALRMGLPVLAVVA
FAQSFGDGVHTSIPAPGLGALGAGRGKDSPLARALAKLGVAADDVAVISKHDTSTLAND
PNETELHERLADALGRSEGAPLFVVSQKSLTGHAAGGAAGVFMHMLCQILRDGVIPPNRS
LDCVDELAGSAHFVWVRDTRLRLGGKFLKAGMLTSLGFGHVSGLVALVHPQAFIASLDP
AQRADYQRRADARLLAGQRRLASAIAAGGAPMYQRPDRRFDHAPERPQEASMLLNPAAR
LGDGEAYIG
```

20. Agora tente listar o conteúdo do arquivo `mt.fasta` utilizando o comando `more` e depois o comando `less`.


```
mury@ubuntu: ~/Downloads/sample
>Rv0001
MTDDPGSGFTTVWNAVVSSELNGDPKVDGPPSSDANLSAPLTPQQRAWNLVQPLTIVEGF
ALLSVPSFVFQNEIERHLRAPITDALSRRLGHQIQLGVRIAPPATDEADTTVPPSENPA
TTSPDTTTTDNDEIDDSAAARGDNQHSWPSYFTERPHNTDSATAGVTSLNRRYTFDTFVIG
ASNRFAHAALAAIAEAPARAYNPLFIWGESGLKTHLLHAAGNYAQRLLFPGMRVYVSTE
EFTNDFINSLRDDRKVAFKRSYRDVLLVDDIQFIEGKEGIEEFFHTFNTLHNANKQI
VISSDRPPKQLATLEDRLRTRFEWGLITDVQPPELETRIAILRKKQMERLAVPDDVLEL
IASSIERNIRELEGALIRVTAFASLNKTPIDKALAEIVLRDLIADANTMQISAATIMAAT
AEYFDTTVEELRGPGKTRALAQSRQIAMYLCRELDLSLPKIGQAFGRDHTTVMYAQRKI
LSEMAERREVFHVKELTTRIRQSKR
>Rv0002
MDAATTRVGLDGLTFRLLRESFADAVSWVAKNLPARPAVPVLSGVLLTGSNDGLTISGFD
YEVSAAEQVGAIEVSPGSLVSGRLLSDITRALPNKPVVDVHVEGNRVALTCGNARFSLPT
MPVEDYPTLPTLPEETGLLPAELFAEASQVAIAAGRDDTLPHLTGIRVEILGETVVLAA
TDRFRLAVRELKWSASSPDI EAAVLVPAKTLAEAAKAGIGGSDVRLSLGTGPGVGKDGLL
GISGNGKRSTTRLLDAEFKFRQLLPTHTAVATMDVAELIEAIKLVALVADRGAVRME
FADGSVRLSAGADDVGRAEEDLVVDYAGEPLTIAFNPTYLTGGLSSLSRSEVSFGFTTAG
KPALLRPVSGDDRPVAGLNGNGPFAVSTDYVYLLMPVRLPG
>Rv0003
MYVRHLGLRDRFSWACVDLELHPGRTVFVGPNGYGKTNLIEALWYSTTLGSHRVSADLPL
IRVGTDRAVISTIVVNDGRECAVDLEIATGRVNKARLNRSVRSTROVVGLRAVLFAPE
DLGLVRGDPADRRRYLDDLAIVRRPAIAAVRAEYERVLQRQTALLKSVPGARYRGDRGVF
DTLEVWDSRLAEHGAELVAARIDLNVQLAPEVKKAYQLLAPESRSASIGYRASMDVTGPS
mt.fasta
```

```
mury@ubuntu: ~/Downloads/sample
mury@ubuntu:~/Downloads/sample$ less mt.fasta
mury@ubuntu:~/Downloads/sample$ more mt.fasta
>Rv0001
MTDDPGSGFTTVWNAVVSSELNGDPKVDGPPSSDANLSAPLTPQQRAWNLVQPLTIVEGF
ALLSVPSFVFQNEIERHLRAPITDALSRRLGHQIQLGVRIAPPATDEADTTVPPSENPA
TTSPDTTTTDNDEIDDSAAARGDNQHSWPSYFTERPHNTDSATAGVTSLNRRYTFDTFVIG
ASNRFAHAALAAIAEAPARAYNPLFIWGESGLKTHLLHAAGNYAQRLLFPGMRVYVSTE
EFTNDFINSLRDDRKVAFKRSYRDVLLVDDIQFIEGKEGIEEFFHTFNTLHNANKQI
VISSDRPPKQLATLEDRLRTRFEWGLITDVQPPELETRIAILRKKQMERLAVPDDVLEL
IASSIERNIRELEGALIRVTAFASLNKTPIDKALAEIVLRDLIADANTMQISAATIMAAT
AEYFDTTVEELRGPGKTRALAQSRQIAMYLCRELDLSLPKIGQAFGRDHTTVMYAQRKI
LSEMAERREVFHVKELTTRIRQSKR
>Rv0002
MDAATTRVGLDGLTFRLLRESFADAVSWVAKNLPARPAVPVLSGVLLTGSNDGLTISGFD
YEVSAAEQVGAIEVSPGSLVSGRLLSDITRALPNKPVVDVHVEGNRVALTCGNARFSLPT
MPVEDYPTLPTLPEETGLLPAELFAEASQVAIAAGRDDTLPHLTGIRVEILGETVVLAA
TDRFRLAVRELKWSASSPDI EAAVLVPAKTLAEAAKAGIGGSDVRLSLGTGPGVGKDGLL
GISGNGKRSTTRLLDAEFKFRQLLPTHTAVATMDVAELIEAIKLVALVADRGAVRME
FADGSVRLSAGADDVGRAEEDLVVDYAGEPLTIAFNPTYLTGGLSSLSRSEVSFGFTTAG
KPALLRPVSGDDRPVAGLNGNGPFAVSTDYVYLLMPVRLPG
>Rv0003
MYVRHLGLRDRFSWACVDLELHPGRTVFVGPNGYGKTNLIEALWYSTTLGSHRVSADLPL
IRVGTDRAVISTIVVNDGRECAVDLEIATGRVNKARLNRSVRSTROVVGLRAVLFAPE
DLGLVRGDPADRRRYLDDLAIVRRPAIAAVRAEYERVLQRQTALLKSVPGARYRGDRGVF
```

21. Outro comando útil são os comandos head e tail que nos permitem listar apenas o início e o final de um arquivo. O parâmetro utilizado para determinar qual a quantidade de linhas a ser exibida é passado com o -n. Experimente: head -n 10 mt.fasta. Depois execute novamente aumentando o tamanho de 10 para outro valor maior/menor.

```

mury@ubuntu:~/Downloads/sample$ head -n 10 mt.fasta
>Rv0001
MTDDPGSGFTTVWNAVSELNQDPKVDDGPSSDANLSAPLTPQQRAWNLVQPLTIVEGF
ALLSVPSSFVQNEIERHLRAPITDALSRRLGHQIQLGVRIAPPATDEADDTTVPPSENPA
TTSPDTTTTDNDEIDDSAAARGDNQHSWPSYFTERPHNTDSATAGVTSLNRRYTFDTFVIG
ASNRFAHAAALAAEAPARAYNPLFIWGESGLGKTHLLHAAGNYAQRLLFPGMRVKYVSTE
EFTNDFINSLRDDRKVAFKRSYRDVDVLLVDDIQFIEGKEGIEEFFHTFNTLHNANKQI
VISSDRPPKQLATLEDRLRTRFEWGLITDVQPPELETRAILRKKQAQMERLAVPDDVLEL
IASSIERNIRELEGALIRVTAFASLNKTPIDKALAEIVLRDLIADANTMQISAATIMAAAT
AEYFDTTVEELRGPGKTRALAQSRQIAMYLCRELTDLSLPKIGQAQFGRDHTTVMYAQRKI
LSEMAERREVFHDHVKELTTRIRQSKR
mury@ubuntu:~/Downloads/sample$ head -n 6 mt.fasta
>Rv0001
MTDDPGSGFTTVWNAVSELNQDPKVDDGPSSDANLSAPLTPQQRAWNLVQPLTIVEGF
ALLSVPSSFVQNEIERHLRAPITDALSRRLGHQIQLGVRIAPPATDEADDTTVPPSENPA
TTSPDTTTTDNDEIDDSAAARGDNQHSWPSYFTERPHNTDSATAGVTSLNRRYTFDTFVIG
ASNRFAHAAALAAEAPARAYNPLFIWGESGLGKTHLLHAAGNYAQRLLFPGMRVKYVSTE
EFTNDFINSLRDDRKVAFKRSYRDVDVLLVDDIQFIEGKEGIEEFFHTFNTLHNANKQI
mury@ubuntu:~/Downloads/sample$

```

22. Outros comando muito utilizados são a cópia, mudança de nome/local de um arquivo, bem como a remoção do mesmo. Para praticarmos, suponha que vamos criar uma copia do arquivo mt.fasta, como mt2.fasta:

```
cp mt.fasta mt2.fasta
```

Na sequencia decidimos mover este arquivo para a pasta pai da pasta sample (nesse caso a pasta Downloads):

```
mv mt2.fasta ../
```

E por último decidimos que o arquivo mt.fasta não era mais necessário e decidimos apagá-lo.

```
rm mt.fasta
```

```

mury@ubuntu:~/Downloads/sample$ cp mt.fasta mt2.fasta
mury@ubuntu:~/Downloads/sample$ mv mt2.fasta ../
mury@ubuntu:~/Downloads/sample$ rm mt.fasta

```

23. Outro comando útil é a criação e remoção de pastas. Suponha que seja necessário criar um pasta para editar uma nova versão do arquivo mt2.fasta. Assim criamos uma pasta com o nome mtv2, por exemplo: mkdir mtv2. Na sequencia movemos o arquivo que está na pasta pai da pasta sample para dentro do diretório criado: mv ../mt2.fasta mtv2.

```

mury@ubuntu:~/Downloads/sample$ mkdir mtv2
mury@ubuntu:~/Downloads/sample$ mv ../mt2.fasta mtv2

```

24. Agora tente remover o diretório mtv2 com o comando rm mtv2 e observe o resultado.


```
mury@ubuntu:~/Downloads/sample$ rm mtv2
rm: cannot remove 'mtv2': Is a directory
```

25. Por último, digite o comando history no terminal e observe o que foi retornado. Para fechar o terminal digite exit.

```
mury@ubuntu:~/Downloads/sample$ history
1  pwd
2  ls
3  ls -c -lt
4  cd Downloads
5  pwd
6  ls
7  clear
8  alias cls='clear/ls'
9  alias cls='clear ls'
10 cls
11 cd ..
12 cls
13 clear
14 clear
15 cls
16 cd Downloads/
17 clear
18 alias cls='clear; ls'
19 cls
20 alias cls='clear; ls'
21 cls
```

Prática de Laboratório – Comandos Linux

1. Um comando que costuma ser muito útil para saber sobre quantidades de texto de um arquivo é o comando `wc` (word count). Ele retorna a quantidade símbolos de 'nova linha' (`\n`), a quantidade de caracteres e de bytes presentes em cada arquivo ou entrada. Acione o comando `wc` sobre o arquivo `sequence.fasta`.

```
mury@ubuntu:~$ cd Downloads/  
mury@ubuntu:~/Downloads$ cd sample/  
mury@ubuntu:~/Downloads/sample$ wc sequence.fasta  
  13  12 681 sequence.fasta  
mury@ubuntu:~/Downloads/sample$
```

2. Compare o resultado acima com o retornado pelo comando: `cat sequence.fasta`

```
mury@ubuntu:~/Downloads/sample$ cat sequence.fasta  
>Cp1002_0126a  
MHFKTRMSLFCTATTAATSLAVASLQPAAVEQPSNTIVSTIMLPTKATVTKFTVSSTKGTARADYSSN  
SITVQPGDTISVKIHSQGGYTEFSELTEFVPSVGR LHTEITFKEGDSGPHPLKVAGWNATSQADRVTFR  
TNDGKPKAITLDTTLEYTYTVGVRATGDPSTRFQLSSSDSNTVFTSASGPKIHVKKTLPSWLSGAFPGA  
FDSLTNLLSPILRALNIL  
>Cp1002_1802  
MLFPSRFQGTFLKPLITAALAVFCVGFTPATAQVIPYTDPDGFYTSIPSAENTTPGTVLSQRDVPMPVLD  
VLVCKMKRIAYTSTHPNGFSTPVTGAVLLPTAPWRGPGRPVALLAPGTQGAGDSCAPSKLLTMGGEYEMF  
SAAALLNRGWTVAVTDYQGLGTPGNHTYMNRKAQGAALLDLGRAITTLNLPDVNNHTPIIPWGYSGGGA  
SAAAAMHRAYAPDVNVVLAYAGGV PANLLSVSSSLEGTALTGALGYVITGMYEITYPEIREPIHNFLNTR  
GQVWLDQTSRDCLPESLLTMPLPDTSLITVSGQRLTSLISDDVFQRAISEQQIGLTAPDIPVFVAQGLND  
GIIPAEQARIMVNGWLSQGADVTYWEDPSALDKLSGHIHVLAASSFLPAVEWAEQRLAALGQPTP
```

3. O comando `wc` consegue abrir arquivos por conta própria, mas pode ser que não se queira abrir um arquivo inteiro, mas apenas parte dele. Por exemplo, suponha que você queira ver as estatísticas apenas sobre a primeira linha do arquivo `sequence.fasta`. Então podemos usar os comandos `head` em um pipeline com o `wc`. Veja o resultado deste comando:

```
mury@ubuntu:~/Downloads/sample$ head -n 1 sequence.fasta | wc  
  1      1     14  
mury@ubuntu:~/Downloads/sample$
```

4. Agora digite apenas `head -n 1 sequence.fasta` e compare com o que foi retornado pelo pipeline.

```
mury@ubuntu:~/Downloads/sample$ head -n 1 sequence.fasta | wc
      1      1     14
mury@ubuntu:~/Downloads/sample$ head -n 1 sequence.fasta
>Cp1002_0126a
mury@ubuntu:~/Downloads/sample$
```

- Qualquer formato textual gerado pelos comandos pode ser direcionada como entrada para o processamento do comando wc. Por exemplo, como saber quantos arquivos existem dentro do diretório proteínas? (sem acento) Vamos lá, tente, na pior das hipóteses você pode baixar os arquivos novamente e recomeçar!

```
mury@ubuntu:~/Downloads/sample$ wc proteínas/
wc: proteínas/: Is a directory
      0      0      0 proteínas/
mury@ubuntu:~/Downloads/sample$
```

- O comando grep é bastante útil porque nos permite a consulta por expressões regulares em vários arquivos. Vamos, por exemplo, considerar o exemplo de sala de aula: Um determinado motivo proteico conservado AKAP foi descrito na literatura como importante para a instauração de uma infecção bacteriana. As proteínas do seu organismo de estudo possuem este motivo? Para resolver esse problema basta digitar o comando abaixo e analisar o resultado:

```
mury@ubuntu:~/Downloads/sample$ grep AKAP proteínas/*
proteínas/Rv0338c: CQSQC PAWNTGKPLSPKLVIMDLRDHWMAKAPYILGQKDASAGGEAGHQEHHPVESGFG
proteínas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAHEWHAMSVQVRKALPSPDEQAPAKAPSPYPARA
proteínas/Rv0540: MSCLPVSVLVAKAPEPGRVKTRLAAAI GDKVAADIAAAALDTLDAAVAAAPVTARAVAl
proteínas/Rv0706: PAKDQSAKSSRRARTEASKAASKVGATAPAKKAAAKAPAKKAPASSGVKKTPAKKAPAK
proteínas/Rv1425: KADVGNQVSSMTASLATHIEDPAKRLAAIHSTLSAKEMAKAPSAHQIMGLTETTPGGL
proteínas/Rv1426c: ATLPTPEMRSRGRNPLRMTAMARRRYVETNNVVCYGPYGRANLADIWRRDLPRDAKAPV
proteínas/Rv1484: MTGLLDGKRILVSGIITDSSIAFHIAARVAQEQAQLVLTGFDRRLRIQRITDRLPAKAPL
proteínas/Rv1530: VAALTEQTGGLADVVDVTAKAPAAFAQAIALARPAGTVVAGTRGVGSGAPGFSPDVVV
proteínas/Rv1985c: GFTAAAAAKAPSLAWNRRDGLQDMLVRKAFRRITRPTHFVPTTEGFTAAARAGLGWGMF
proteínas/Rv2164c: QTSFMLSPFDRPAPAKNTSQAKARAKARKAKAPKLVRPTPMERLAARLTSIDLRPRTLAN
proteínas/Rv2215: APYVTPLVRLASENNIDLAVGTGTGVGGIRKQDVLAAAEQKKRAKAPPAAQAAAAAPA
proteínas/Rv2448c: DFLAKAPDAVIAKIRDQRVAQQETERITRLAALQ
proteínas/Rv2536: VFLVGIVGAVGRWLVDRLAKAPVRHHGLAAEHERAADTDVFSAVRADDSPGTGEMQVAQ
proteínas/Rv2703: ASAPQDTTSTIPKRKTRAAKSAAAKAPSARGHATKPRAPKDAQHEAATDPEDALDSVE
proteínas/Rv2780: VPGAKAPKLVSNSLVAHMKPGAVLVDIAIDQGGCFEGSRPTTYDHPTFAVHDTLFYCVAN
proteínas/Rv2973c: LGMVVVDEQHRFGVEQRDQLRAKAPAGITPHLLVMTATPIPRTVALTIVYGDLETSTLREL
proteínas/Rv3099c: LDEIAATMGFVVGKLAKAPDGSRVLLLELTGPLSRISVSDGRARVDDFGGPAPTATIR
proteínas/Rv3269: MAIQVFLAKATTITVITGLAGVTAYEILKAAAKAPLRTAVSAAALGLRGTRKAEAAES
proteínas/Rv3456c: VTSEANRARRVAAQAKAKKAAAMPTEESEAKPAEEGDVVGASEPDAKAPEEPPAEAPEN
proteínas/Rv3691: LAKAPGDLLVAPTSTRRTALTPQLRIAAASPFPNSQPNCTLREANRAGSVQWGPSDTYQA
proteínas/Rv3808c: YEALKNTDCQQILFMDDDIRLEPDSILRVLAMHRFAKAPMLVGGQMLNLQEPSHLHIMGE
mury@ubuntu:~/Downloads/sample$
```

- O comando grep também possibilita fazer a contagem da quantidade de resultados retornados. Ao invés de você ficar contando na tela a quantidade de motivos retornados (destacados em vermelho) é muito mais prático contá-los com o uso do próprio grep. Acesse o manual do grep e descubra qual parâmetro o configura para fazer a contagem, ao invés de listar o motivo buscado na tela. Em seguida: Use este parâmetro no último comando linux para contar quantos motivos existem.

```
mury@ubuntu:~/Downloads/sample$ grep -c "$AKAP" proteinas/*
proteinas/Rv0001:10
proteinas/Rv0002:8
proteinas/Rv0003:8
proteinas/Rv0004:5
proteinas/Rv0005:13
proteinas/Rv0006:15
proteinas/Rv0007:7
proteinas/Rv0008c:4
proteinas/Rv0009:5
proteinas/Rv0010c:4
proteinas/Rv0011c:3
proteinas/Rv0012:6
proteinas/Rv0013:5
proteinas/Rv0014c:12
proteinas/Rv0015c:9
proteinas/Rv0016c:10
proteinas/Rv0017c:9
proteinas/Rv0018c:10
proteinas/Rv0019c:4
proteinas/Rv0020c:10
proteinas/Rv0021c:7
proteinas/Rv0022c:4
```

8. O comando `grep '>' mt.fasta` apenas lista todas as linhas que possuem o sinal de maior. Verifique executando este comando. Perceba que não é isso o que queremos. Dessa forma utilize o parâmetro que você descobriu no item anterior para contar quantas sequencias existem dentro do arquivo. Compare o resultado com o retornado pelo item 2.3.

```
mury@ubuntu:~/Downloads/sample/mtv2$ grep -c '>' mt2.fasta
3988
```

9. Encontre os arquivos modificados a mais de dois dias dentro do diretório sample com o comando `find . -mtime +2`

```
mury@ubuntu: ~/Downloads/sample
mury@ubuntu:~/Downloads/sample$ find . -mtime +2
./andromeda.jpg
./SOAPdenovo2.pdf
./sequence.fasta
./proteinas
./proteinas/Rv0296c
./proteinas/Rv1283c
./proteinas/Rv0054
./proteinas/Rv3143
./proteinas/Rv0529
./proteinas/Rv3188
./proteinas/Rv1882c
./proteinas/Rv0408
./proteinas/Rv2675c
./proteinas/Rv2866
./proteinas/Rv3237c
./proteinas/Rv1206
./proteinas/Rv3122
./proteinas/Rv1430
./proteinas/Rv2737A
./proteinas/Rv2337c
./proteinas/Rv0868c
./proteinas/Rv0010c
./proteinas/Rv0257
```

10. Encontre os arquivos modificados a menos de um dia dentro do diretório sample com o comando `find . -mtime -1`

```
mury@ubuntu:~/Downloads/sample$ find . -mtime -1
.
./mtv2
./mtv2/mt2.fasta
./cat
mury@ubuntu:~/Downloads/sample$
```

11. Encontre os arquivos filhos que possuem um tamanho maior que 500 kilobytes com o comando `find . -size +500k`

```
mury@ubuntu:~/Downloads/sample$ find . -size +500k
./mtv2/mt2.fasta
./Poster.pdf
mury@ubuntu:~/Downloads/sample$
```

12. Encontre os arquivos filhos que possuem um tamanho maior que 500 kilobytes e liste as propriedades básicas destes arquivos com o comando `find . -size +500k -ls`

```
mury@ubuntu:~/Downloads/sample$ find . -size +500k -ls
524335 1364 -rw-rw-r-- 1 mury mury 1394290 Jun 20 14:35 ./mtv2/mt2.fasta
525220 600 -rw-rw-r-- 1 mury mury 611453 May 5 2014 ./Poster.pdf
mury@ubuntu:~/Downloads/sample$
```

13. Encontre todos os arquivos cujo nome termine com pdf com o comando `find . -name '*.pdf' -ls`

```
mury@ubuntu:~/Downloads/sample$ find . -name '*.pdf' -ls
525211 376 -rw-rw-r-- 1 mury mury 382965 Jan 7 2014 ./SOAPdenovo2.pdf
525220 600 -rw-rw-r-- 1 mury mury 611453 May 5 2014 ./Poster.pdf
525221 316 -rw-rw-r-- 1 mury mury 322808 Feb 24 2014 ./pan-genome-size.pdf
mury@ubuntu:~/Downloads/sample$
```

14. Encontre arquivos ou diretórios, a partir do diretório sample, que possuem a permissão de execução para o grupo 'outros'. Lembre-se que existem os grupos de permissões: usuário proprietário, grupo proprietário e outros, que utilizam as letras 'u', 'g' e 'o' para significar cada grupo, respectivamente, sendo que as permissões podem ser r=read, w=write e x=executar. Este comando gera o relatório: `find . -perm -o=x`

```
mury@ubuntu:~/Downloads/sample$ find . -perm -o=x
.
./mtv2
./proteinas
mury@ubuntu:~/Downloads/sample$
```

15. Encontre os diretórios a partir da pasta sample que possuem permissão de escrita para o grupo proprietário: `find . -perm -g=w -type d`

```
mury@ubuntu:~/Downloads/sample$ find . -perm -g=w -type d
.
./mtv2
./proteinas
mury@ubuntu:~/Downloads/sample$
```

16. Encontre e liste os arquivos de proteínas, da pasta proteínas, apenas com arquivos cujas sequências de proteínas forem maiores do que 1 kilobyte (1k): `find ./proteinas -size +1k -type f -ls`

```
./proteinas
mury@ubuntu:~/Downloads/sample$ find ./proteinas -size +1k -type f -ls
533480 4 -rw-rw-r-- 1 mury mury 1167 May 5 2014 ./proteinas/Rv2566
534931 8 -rw-rw-r-- 1 mury mury 4230 May 5 2014 ./proteinas/Rv2048c
534318 4 -rw-rw-r-- 1 mury mury 1122 May 5 2014 ./proteinas/Rv0630c
534185 4 -rw-rw-r-- 1 mury mury 3364 May 5 2014 ./proteinas/Rv0355c
533297 4 -rw-rw-r-- 1 mury mury 1038 May 5 2014 ./proteinas/Rv2101
535728 4 -rw-rw-r-- 1 mury mury 3787 May 5 2014 ./proteinas/Rv3350c
535740 4 -rw-rw-r-- 1 mury mury 1119 May 5 2014 ./proteinas/Rv3370c
532677 4 -rw-rw-r-- 1 mury mury 1263 May 5 2014 ./proteinas/Rv1020
535231 4 -rw-rw-r-- 1 mury mury 3130 May 5 2014 ./proteinas/Rv2524c
533612 4 -rw-rw-r-- 1 mury mury 2233 May 5 2014 ./proteinas/Rv2933
535587 4 -rw-rw-r-- 1 mury mury 1138 May 5 2014 ./proteinas/Rv3080c
534674 4 -rw-rw-r-- 1 mury mury 1361 May 5 2014 ./proteinas/Rv1450c
535726 4 -rw-rw-r-- 1 mury mury 3219 May 5 2014 ./proteinas/Rv3347c
534134 4 -rw-rw-r-- 1 mury mury 1433 May 5 2014 ./proteinas/Rv0236c
534749 4 -rw-rw-r-- 1 mury mury 1201 May 5 2014 ./proteinas/Rv1640c
534941 4 -rw-rw-r-- 1 mury mury 1223 May 5 2014 ./proteinas/Rv2062c
534226 4 -rw-rw-r-- 1 mury mury 1574 May 5 2014 ./proteinas/Rv0425c
532308 4 -rw-rw-r-- 1 mury mury 1112 May 5 2014 ./proteinas/Rv0386
533780 4 -rw-rw-r-- 1 mury mury 1547 May 5 2014 ./proteinas/Rv3296
534593 4 -rw-rw-r-- 1 mury mury 1167 May 5 2014 ./proteinas/Rv1251c
534044 4 -rw-rw-r-- 1 mury mury 1212 May 5 2014 ./proteinas/Rv3910
534751 4 -rw-rw-r-- 1 mury mury 1037 May 5 2014 ./proteinas/Rv1651c
```

17. Crie um arquivo multi-fasta denominado maiores.fasta, a partir da pasta proteínas, contendo apenas os arquivos com sequências de proteínas que forem maiores do que 1 kilobyte (1k): `find ./proteinas -size +1k -type f -exec cat > maiores.fasta {} \;`

```
mury@ubuntu:~/Downloads/sample$ find ./proteinas -size +1k -type f -exec cat > maiores.fasta {} \;
mury@ubuntu:~/Downloads/sample$
```

18. Encontre os mesmos arquivos do item anterior, mas ao invés criar um arquivo com as sequências, execute um comando `grep` para procurar pelo motivo conservado AKAP: `find ./proteinas -size +1k -type f -ls -exec grep AKAP {} \;`

```
mury@ubuntu:~/Downloads/sample$ find ./proteinas -size +1k -type f -exec cat > maiores.fasta {} \;
mury@ubuntu:~/Downloads/sample$ find ./proteinas -size +1k -type f -ls -exec grep AKAP {} \;
533480 4 -rw-rw-r-- 1 mury mury 1167 May 5 2014 ./proteinas/Rv2566
534931 8 -rw-rw-r-- 1 mury mury 4230 May 5 2014 ./proteinas/Rv2048c
534318 4 -rw-rw-r-- 1 mury mury 1122 May 5 2014 ./proteinas/Rv0630c
534185 4 -rw-rw-r-- 1 mury mury 3364 May 5 2014 ./proteinas/Rv0355c
533297 4 -rw-rw-r-- 1 mury mury 1038 May 5 2014 ./proteinas/Rv2101
535728 4 -rw-rw-r-- 1 mury mury 3787 May 5 2014 ./proteinas/Rv3350c
535740 4 -rw-rw-r-- 1 mury mury 1119 May 5 2014 ./proteinas/Rv3370c
532677 4 -rw-rw-r-- 1 mury mury 1263 May 5 2014 ./proteinas/Rv1020
535231 4 -rw-rw-r-- 1 mury mury 3130 May 5 2014 ./proteinas/Rv2524c
533612 4 -rw-rw-r-- 1 mury mury 2233 May 5 2014 ./proteinas/Rv2933
535587 4 -rw-rw-r-- 1 mury mury 1138 May 5 2014 ./proteinas/Rv3080c
534674 4 -rw-rw-r-- 1 mury mury 1361 May 5 2014 ./proteinas/Rv1450c
535726 4 -rw-rw-r-- 1 mury mury 3219 May 5 2014 ./proteinas/Rv3347c
534134 4 -rw-rw-r-- 1 mury mury 1433 May 5 2014 ./proteinas/Rv0236c
534749 4 -rw-rw-r-- 1 mury mury 1201 May 5 2014 ./proteinas/Rv1640c
534941 4 -rw-rw-r-- 1 mury mury 1223 May 5 2014 ./proteinas/Rv2062c
534226 4 -rw-rw-r-- 1 mury mury 1574 May 5 2014 ./proteinas/Rv0425c
AAGLLSGYLLARKVVDQAQRPAPAHWHAMSVQVRKALPSPDEQAPAKAPSPSPYPARA
532308 4 -rw-rw-r-- 1 mury mury 1112 May 5 2014 ./proteinas/Rv0386
533780 4 -rw-rw-r-- 1 mury mury 1547 May 5 2014 ./proteinas/Rv3296
```


19. O comando sed nos auxilia realizando edições em lote em diversos arquivos. Vejamos um exemplo simples de uso do sed, os exemplos mais interessantes são criados quando conjugamos o sed com expressões regulares. Suponha que um novo motivo proteico está sendo estudado, o motivo ACP. Deseja-se editar o arquivo maiores.fasta, criado nos item anterior, para delimitar todas as ocorrências de ACP com um tipo de tag, de modo a que este motivo não fique oculto em possíveis análises visuais do arquivo. Para tanto onde houver "ACP" deve ser substituído por "< ACP >". Execute este comando para realizar tal operação: sed -i "s/ACP/< ACP >/g" maiores.fasta

```
mury@ubuntu:~/Downloads/sample$ sed -i "s/ACP/< ACP >/g" maiores.fasta
mury@ubuntu:~/Downloads/sample$
```

20. Os comandos grep e sed se tornam muito poderosos quando expressões regulares são utilizadas nos parâmetros de busca. Um exemplo simples pode ser a busca de um motivo conservado no qual existe um aminoácido que pode variar entre n letras, digamos n=3 letras. Suponha o motivo AKAP no qual a segunda letra A pode ser substituída pelos aminoácidos R e P. Este comando gera o relatório que precisamos: grep "AK[ARP]P" proteínas/*

```
mury@ubuntu:~/Downloads/sample$ grep "AK[ARP]P" proteínas/*
proteínas/Rv0338c:CQSQCPAWNTGKPLSPKLVIMDLRDHWMAKAPYILGQKDASAGGEAGHQEHHPESGFG
proteínas/Rv0338c:MAAGAKRPGAKKAAPTAAAPAAAPVKGLGIAAGAKRPGAKKTPPPAPGLAEPAAQPP
proteínas/Rv0425c:AAGLLSGYLLARKVVDAQAPRPAPAHWEHMSVEQVRKALPSPDEQAPAKAPSPYPARA
proteínas/Rv0540:MSCLPVSVLVAKAPEGRVKTRLAAAIQDKVAADIAAAALLDTLDAVAAAPVTARAV
proteínas/Rv0573c:DTGYSYPVAVSDRIVGELARLRHADTAEAHPGSNVVGAKAKRP
proteínas/Rv0581:MDKTTVYLPDELKAAVKRAARQGVSEAQVIREIRAAGVAKPPPRGGLYAGSEPIARR
proteínas/Rv0586:TLAALTTMTAAAKRPSDYRKLADAICSGDPTGAKKAAQDLLELANTSLMAVLVSQASRQ
proteínas/Rv0706:PAKQDSAKSSRRARTEASKAASKVGATAPAKKAAAKAPAKKAPASSGVKKTPAKKAPAK
proteínas/Rv0949:MSVHATDAKPPGPSPADQLDGLNPQQRQAVVHEGSPLLIVAGAGSGKTAVLTRRIAYLM
proteínas/Rv0983:GSGIILSAEGLILTNHVIAKPPLGSPPPKTTVTFSDGRTAPFTVVGADPTSDIAVV
proteínas/Rv1024:MPEAKRPESKRRSPASRPKGAGDSVRGGGRATKPSAKPSTPAPHASRKTTRTPHEHIVEPI
proteínas/Rv1215c:MARNPSPALDRPWRRPGALRYALERVGVAKPPITVTDPPADVIERDVEVPTRDGTLRL
proteínas/Rv1425:KADVGNQVSSMTASLATHIEDPAKRLAAIHESLTSAKEMAKAPSAHQIMGLTETTPPGLL
proteínas/Rv1426c:ATLPTPMRSRGRNLPLRTAMARRRYVETTNVVCYGPYGRANLADIWRRRDLPRDAKAPV
proteínas/Rv1483:MTATATEGAKPPFVFSRSLVTGGNRGIGLAIAQRLAADGHKVAVTHRGSGAPKGLFGVEC
proteínas/Rv1484:MTGLLDGKRILVSGIITDSSIAFHIAQVAQEQAQLVLTGFDRLLIQRITDRLPAKAPL
proteínas/Rv1530:VAALTEQTGGLADVVDVTAKAPAAFAQAIALARPAGTVVVAGTRGVGSGAPGFSPPDVVV
proteínas/Rv1660:TWRSLEIGNLSSASVLHVLRTIAKPPPSGSPGLMIAMGPFCSELVLLRWH
proteínas/Rv1731:TDWAKRPVIERAAVIRRYRDLVIENREFLMDLLQAEAGKARWAAQEEIVDLIANANYAR
proteínas/Rv1985c:GFTAAAAAKAPSLAWNRRDGLQDMLVRKAFRAITRPTHEVPTTEGFTAAARAGLGWGMF
```

21. Outra variação do comando anterior seria a possibilidade de que qualquer outro aminoácido pudesse entrar no lugar da segunda letra A. Neste caso o ponto '.' vem ao nosso favor: grep "AK.P" proteínas/*

```
mury@ubuntu:~/Downloads/sample$ grep "AK.P" proteinas/*
proteinas/Rv0014c: EGLSADLDVAVLKALAKNPENRYQTAEMRADLVRVHNGEPPEAPKVLTDARTSLLSSA
proteinas/Rv0060: ELLASTHWATREGAKFATAAAVRKWKTKRGRIYSDDRIGVALDRILMTA
proteinas/Rv0061: LLAVAHQVAKTPSATRAIAFRHVRLMRVWICAGNRGRKHRRCTTQYRSTQASKLQLH
proteinas/Rv0072: PRVSEGRSPSKPDEVAASSTMGRHLGDTVEVGARRLRVVGIVPNSTALAKIFNVFLTTEG
proteinas/Rv0126: ERDVMYAEYAKDPRMKANVGIIRRLAPLNDNRNQIELFTALLSLPGSPVLYYGDEIGM
proteinas/Rv0147: AVRERVIREVPAGGMVNHLAFQVSTAKLPFGGVGASGMGAYHGRWGEFEFSHRKSVLTK
proteinas/Rv0156: LVLQVILFVAVVFGTLNVIGGFIVTDRMLGMFKAKKPAVPAKPDREALR
proteinas/Rv0197: RVVLNEIAKDPGRSMIVIDPVVTDTKMADFHLRVQPGCDACLAAALVAVLVQENLCNEA
proteinas/Rv0206c: VGGTPALELDSIHGLF AKMPLMVILLTTTIVLMFLAFGSVVLPIKATLMSALTGSTMG
proteinas/Rv0213c: PGTPLYQAYSAGYLTAKMPLGQWFEFVDPASRVYADVAVKVPDVGISFDEAEAYFLSR
proteinas/Rv0283: SSVALKDSGKTIISDTVQYAVLPDGLQGISPVLAAILRNNSYGLQPPRLGADEVAKLP
proteinas/Rv0309: PNHWMSGDDNSPTFNSMQVCQKSCQCFSTADSENLIQPYKHSVMGVNKAQKVPKGSAF
proteinas/Rv0338c: CQSQCPAWNTGKPLSPKLVIDLRDHWMAKAPYILGQKDSAGGEAGHQEHVHPESGFG
proteinas/Rv0338c: MAAGAKRP GAKKAAPTAPAAPAAPVKGLGIAAGAKRP GAKKTTPAPGLAEPAAQPPQ
proteinas/Rv0353: MAKNPKGESRTFLISVAAELAGMHAQTLRTYDRLGLVSPRSTSGGRRYSLHDVLLRQ
proteinas/Rv0363c: SGLGVKDMVTGAVALAEFTHVIAAKYPVNVALHTDHPCKDKLSYVRPLLAISAQRVSKG
proteinas/Rv0370c: GVAETIDHVAALVALGVADLTAAOSSPALASLGALAKTDDRTQIRDAYQAFTECSHA
proteinas/Rv0372c: LVASTVRGASILDLSLDAERARVHTPVCLATIGAKTAEIIVSIAELIATLRGGCPRG
proteinas/Rv0373c: TGFARDYIMVGEIAANRDGKILAIRSNVLADHGAFNAQAAPAKYPAGFFGVFTGSYDIEA
proteinas/Rv0375c: CTVLDAVCLAKGPSGEREIAIDDFLVGPYETALAHNEVLEVRIPLRHNTSSAYAKVERR
proteinas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAHEWHAMSVEQVRKALPSPDEQAPAKAPSPYPARA
proteinas/Rv0444c: AEQVLTAQDVRTVSRPLGAGTATVVSORDRTGLVMNNVAPPSTRTVYQWLLGGAKGP
```

22. Mais uma variação do comando inicial. Suponha que um outro motivo possua uma ou mais letras A no local onde ocorre a segunda letra A. Neste caso o operador de repetição de ER's resolve a nossa consulta. Nessa caso o operador '+' precisa ser precedido por uma barra: grep "AKA\+P" proteinas/*

```
mury@ubuntu:~/Downloads/sample$ grep "AKA\+P" proteinas/*
proteinas/Rv0338c: CQSQCPAWNTGKPLSPKLVIDLRDHWMAKAPYILGQKDSAGGEAGHQEHVHPESGFG
proteinas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAHEWHAMSVEQVRKALPSPDEQAPAKAPSPYPARA
proteinas/Rv0540: MSCLPVSVLVVAKAPEGRVKTRLAAAIIGDKVAADIAAAALLDLDVAAAAPVTARAVL
proteinas/Rv0706: PAKDQRSASSRRARTEASKAASKVGATAPAKKAAAKAPAKKAPASSGVKKTPAKKAPAK
proteinas/Rv0710: MMAEAKTGAKAAPRVAKAKAAAPKKAAPNDAAEIGAANAANVGPKHTPRTPKPRGRRKT
proteinas/Rv1425: KADVGNQVSSMTASLATHIEDPAKRLAATHESLSAKEMAKAPSAHQIMGLTETTPPGLL
proteinas/Rv1426c: ATLPTPEMRSRGNLPLRTAMARRRYVETTNVVCYGPYGRANLADIWRRRDLPRDAAKAPV
proteinas/Rv1484: MTGLLDGKRILVSGIITDSSIAFHIAARVAQEQAQLVLTGFDRLRLIQRITDRLPAKAPL
proteinas/Rv1530: VAALTEQTGGLADVVDVTAKAPAAFAQAIALARPAGTVVAVAGTRGVGSGAPGFSPDVVV
proteinas/Rv1657: MSRAKAAPVAGPEVAANRAGQARIVAILSSAQVRSQNELAALLAAEGIEVTQATLSRDL
proteinas/Rv1985c: GFTAAAAAKAPSLAWNRRDGLQDMLVRKAFFRAITRPTHFVPTTEGFTAAARAGLGWGMF
proteinas/Rv2164c: QTSPLSPFDRPAPAKNTSQAKARAKARKAKAPKLVRPMPMERLAARLTSIDLRPRTLAN
proteinas/Rv2215: APYVTPLVRLASENNIDLAGVTGTGVGGRIRKQDVLAAAEQKKRAKAPAPAAQAAAAPA
proteinas/Rv2448c: DFLAKAPDAVIAKIRDRQVAAQQUETERITTRLAALQ
proteinas/Rv2536: VFLVGIVGVAVGRWLVDRLAKAPVRHHGLAAEHERAADTVFSAVRADDSPTEGEMQVAQ
proteinas/Rv2703: ASAPQDTTSTIPKRKTRAAKSAAAKAPSARGHATKPRAPKDAQHEAATDPEDALDSVE
proteinas/Rv2780: VPGAKAPKLVSNLSVAHMKPGAVLVDIAIDQGGCFEGSRPTTYDHTFAVHDTLFYCVAN
proteinas/Rv2973c: LGMVVVDEQHRFGVEQRDQLRAKAPAGITPHLLVMTATPIPTVALTVYGDLETSTLREL
proteinas/Rv3014c: DEGELFALTERDLLRTDLFRTKAGEL SANGKRLLVNLDKAKAAPLWRVLVALSIRHVGP
proteinas/Rv3099c: LDEIAATMGFVVGKLAKAPDGSRLLELTGPLSRIRVSDGRARVDDFGGPAPTATIR
proteinas/Rv3144c: TESARAPEAASAPPEAVVEVPELEVPAMGVLPVDPKVAKAAPLSTTRVGQSAGSGIP
```

23. Uma variação do último comando. Suponha que no lugar do segundo A possa existir zero ou mais ocorrências de A: grep "AKA*A" proteinas/*

```

mury@ubuntu:~/Downloads/sample$ grep "AKA*P" proteinas/*
proteinas/Rv0006: LIELLIDIEIQAAILDMQLRLAALERQRIIDDLAKIEAIEDLILAKPERQRGIVR
proteinas/Rv0033: TVGDVYTSVAVWFPEAKAPLPGKGA
proteinas/Rv0038: ETAVYNVLPQWAKLAKPKTMFIGGPVKRDAALCLAVLRVGADPEGVPLRHVAGRLVMV
proteinas/Rv0048c: TSDPGAKPDGIAPVLTTPRQLHSLGGLTGLLEQTRKRFQDTMGYRLVIYPEASLDRVD
proteinas/Rv0090: DLPVLRAGFKVDQMTVIHALEKALAAPKSTLALITGMLAAYAVLQAVEGVGLWLKRWG
proteinas/Rv0095c: RITKPEAGRRSAEAKP
proteinas/Rv0102: FVSVAEPVGFFAASLAGALCLGALIHVMTAKPEPDGLIDAAAFRIHLAERVSGLWLGL
proteinas/Rv0103c: EAKPVRAYPAAASVVGTVVMDGRLVIEATAVGADTQFAAMVRLVEQAQTQKARAQLADH
proteinas/Rv0130: GTVQATVSTTVEVFGSAKPAACVAESIVRYVA
proteinas/Rv0156: LVLQVILFVAVVFGTLNVIGGFIVTDRMLGMFKAKKPAVPKAPDRDEALR
proteinas/Rv0165c: MIKHDVVMTLWPERPNNKPPSPRQVPGNPGPTLKVLAHVNAPLSAKPRSQPLPRAQ
proteinas/Rv0171: MRTLEPPNMRMIGLGIIVVALLVAVGQSFTSVPMLFAKPSYYGQFDSGGLHKGDRVRI
proteinas/Rv0189c: MPQTTDEAASVSTVADIKPRSRDVTGDKAAARGMLRAVGMDDEFKAPQIGVASSWNE
proteinas/Rv0192: AQODPTSFVGPPFPPTFNPVDGAMVGVAKPIVINFAPVIADRAMAESATHISSIPPVP
proteinas/Rv0199: ATSVSENAGAKPQTVHWNLRLDVSDVDGKLMISRLESIR
proteinas/Rv0207c: GFAVFAKPKVDESDVDVDRMLAHIDERYREGLAALVVASADQAFRQPLEAVARSGTPVQ
proteinas/Rv0242c: VVGCTPEAAASTNERIAQRALEGFTRSLGKELRRGATTALVYLSPDAKPAATGLESTMR
proteinas/Rv0269c: LLMLEELCPQQAQSAPKLEIARAKTRAEAMAALDIHRDRYPGAALLRPADVLDGM
proteinas/Rv0291: GLNATEVVRRLTATAHRGARESSNIVGAGNLDAVALTWQLPAEPGGGAAPKPVADPPV
proteinas/Rv0338c: CQSQCPAWNTGKPLSPKLVIDLRDHWMAKAPYILGQKDSAGGEAGHQEHHVPESGFG
proteinas/Rv0338c: EAKPQPEPAAPPKPTDGDPAAPAPVKGLGIARGARPPGKR
proteinas/Rv0357c: QRDLCRAKPVYEELPGWEDISGAREFDDLPKARDYVLRLEQLAGAPVSCIGVGPGRQ

```

24. Liste os cabeçalhos dos arquivos presentes no diretório proteinas com o comando: `grep ">" proteinas/*`

```

mury@ubuntu: ~/Downloads/sample
proteinas/Rv0001: >Rv0001
proteinas/Rv0002: >Rv0002
proteinas/Rv0003: >Rv0003
proteinas/Rv0004: >Rv0004
proteinas/Rv0005: >Rv0005
proteinas/Rv0006: >Rv0006
proteinas/Rv0007: >Rv0007
proteinas/Rv0008c: >Rv0008c
proteinas/Rv0009: >Rv0009
proteinas/Rv0010c: >Rv0010c
proteinas/Rv0011c: >Rv0011c
proteinas/Rv0012: >Rv0012
proteinas/Rv0013: >Rv0013
proteinas/Rv0014c: >Rv0014c
proteinas/Rv0015c: >Rv0015c
proteinas/Rv0016c: >Rv0016c
proteinas/Rv0017c: >Rv0017c
proteinas/Rv0018c: >Rv0018c
proteinas/Rv0019c: >Rv0019c

```

25. Modifique o cabeçalho de todos os arquivos dentro da pasta proteinas para incluir o prefixo Mt assim adaptar os arquivos para utilização com o tal programa: `sed -i "s/> \(Rv[0-9]\)+c(.*?)>/>Mt_1/g" proteinas/*`

```
mury@ubuntu: ~/Downloads/sample
proteinas/Rv3923c:>Rv3923c
proteinas/Rv3924c:>Rv3924c
mury@ubuntu:~/Downloads/sample$ sed -i "s/>\(Rv[0-9]\|c\|?\)/>Mt_\1/g" proteinas/*
mury@ubuntu:~/Downloads/sample$ grep ">" proteinas/*
proteinas/Rv0001:>Mt_Rv0001
proteinas/Rv0002:>Mt_Rv0002
proteinas/Rv0003:>Mt_Rv0003
proteinas/Rv0004:>Mt_Rv0004
proteinas/Rv0005:>Mt_Rv0005
proteinas/Rv0006:>Mt_Rv0006
proteinas/Rv0007:>Mt_Rv0007
proteinas/Rv0008c:>Mt_Rv0008c
proteinas/Rv0009:>Mt_Rv0009
proteinas/Rv0010c:>Mt_Rv0010c
proteinas/Rv0011c:>Mt_Rv0011c
proteinas/Rv0012:>Mt_Rv0012
proteinas/Rv0013:>Mt_Rv0013
proteinas/Rv0014c:>Mt_Rv0014c
proteinas/Rv0015c:>Mt_Rv0015c
proteinas/Rv0016c:>Mt_Rv0016c
proteinas/Rv0017c:>Mt_Rv0017c
proteinas/Rv0018c:>Mt_Rv0018c
proteinas/Rv0019c:>Mt_Rv0019c
proteinas/Rv0020c:>Mt_Rv0020c
proteinas/Rv0021c:>Mt_Rv0021c
```