

T01 – SO Linux

Igor Augusto Reis Gomes – 12011BSI290 – igor.augusto@ufu.br

Heitor Guimarães Da Fonseca Filho – 12011BSI203 – heitor.filho@ufu.br

*Fizemos apenas os comando avançados dado que já somos usuários de linux (Ubuntu) nativo.

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.

```
~/Downloads
> cd sample

~/Downloads/sample
> ls
andromeda.jpg      Poster.pdf  sequence.fasta
pan-genome-size.pdf proteínas    SOAPdenovo2.pdf

~/Downloads/sample
> wc sequence.fasta
13 12 681 sequence.fasta
```

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

```
~/Downloads/sample
> cat sequence.fasta
> Cp1002_0126a
MHFKTRMSLFCTATTAATSLAVASLQPAAAVEQPSNTIVSTIMLPKATVTKTFTVSSTKGTARADYSSN
SITVQPGDTISVKIHSQGGYTEFSELTEFVPSVGRRLHTESITFKEGDSGPHPLKVAGWNATSQADRVTFR
TNDGKPKAITLDTTLEYTYTVGVRATGDPSTRFQLSSSDSNTVFTSASGPKIHVKKTLPSWLSGAFPGAI
FDSLTLNLLSPILRALNIL
> Cp1002_1802
MLFPSRFQGTFLKPLITAALAVFCVGFTPATAQVIPYTDPDGFYTSIPSAENTTPGTVLSQRDVPMPVLD
VLVKMKRIAYTSTHPNGFSTPVTGAVLLPTAPWRGPGPRPVALLAPGTQAGDSCAPSKLLTMGGEYEMF
SAAALLNRGWTVAVTDYQGLGTPGNHTYMNRKAQGAALLDLGRAITTLNLPDVNNHTPIIPWGYSQGGGA
SAAAAEMHRAYPADPNVVLAYAGGVPANLLSVSSSLEGTALTGALGVVITGMYEIYPEIREPIHNFNTR
QQVWLDQTSRDCLPESLLTMPLPDTSLTVSGQRLTSLISDDVFQRAISEQQIGLTAPDIPVFVAQGLND
GIIPAEQARIMVNGWLSQGADVTYWEDPSPALDKLSGHIHVLASSFLPAVEWAEQRLAALGQPTP
```

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:

```
~/Downloads/sample
> head -n 1 sequence.fasta | wc
1      1      14
```

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

```
~/Downloads/sample
> head -n 1 sequence.fasta
> Cp1002_0126a
```

5. 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!

```
~/Downloads/sample
> wc proteínas/
wc: proteínas/: Is a directory
      0      0      0 proteínas/
```

6. 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:

```
~/Downloads/sample
> grep AKAP proteínas/*
proteínas/Rv0338c:CQSQCPA WNTGKPLSPKLVIMDLRDHWMAKAPYILGQK DASAGGEAGHQEHHPVESGFG
proteínas/Rv0425c:AAGLLSGYLLARKVVDAQAPRPAPAHEWHAMSVEQVRKALPSPDEQAPAKAPSPSPYPARA
proteínas/Rv0540:MSCLPVSVLVVAKAPEPGRVKTRLAAAIGDKVAADIAAAALLDTLDAAVAAAPVTARAVAL
proteínas/Rv0706:PAKDQSAKSSRRARTEASKAASKVGATAPAKKAAAKAPAKKAPASSGVKKTTPAKKAPAK
proteínas/Rv1425:KADVGNQVSSMTASLATHIEDPAKRLAAIHSTLSAKEMAKAPSAHQIMGLTETTPGLL
proteínas/Rv1426c:ATLPTPMRSRGRNPLRTAMARRRYVETTNVVCYGPYGRANLADIWRRDLPRDAKAPV
proteínas/Rv1484:MTGLDGRILVSGIITDSSIAFHIA RVAQEQAQLVLTGFDRLLIQRITDRAKAPL
proteínas/Rv1530:VAALTEQTGGLADVVDVTAKAPAFAQAIALARPAGTVVAVAGTRGVGSGAPGFSDDVVV
proteínas/Rv1985c:GFTAAAAKAPSLAWNRRDGLQDMLVRKAFRRITRPTHFVPTTEGFTAAARAGLWGGMF
proteínas/Rv2164c:QTSPLSPFDRPAPAKNTSOAKARAKARKAKAPKLVRPTPMERLAARLTSIDLPRPTLAN
proteínas/Rv2215:APYVTPLVRLASENNIDLAGVTGTGVGGRIRKQDVLAAAEQKKAKAPAPAAQAAAAAPA
proteínas/Rv2448c:DFLAKAPDAVIAKIRDRQRVAQQETERITTRLAALQ
proteínas/Rv2536:VFLVGIVGAVGRWLVDRLAKAPVRHHGLAAEHERAADTDVFSAVRADDSPTEGEMQVAQ
proteínas/Rv2703:ASAPQDTTSTIPKRKTRAAAKSAAAKAPSARGHATKPRAPKDAQHEAATDPEDALDSVE
proteínas/Rv2780:VPGAKAPKLVSNSLVAHMKPGAVLVDIAIDQGGCFEGSRPTTYDHPFVAVHDTLFYCVAN
proteínas/Rv2973c:LGMVVVDEQHRFGVEQRDQLAKAPAGITPHLLVMTATPIPTVALTVYGDLETSTLREL
proteínas/Rv3099c:LDEIAATMGFVVGKLAKAPDGSRVLLLTGPLSRISRVSDGRARVVDGFGGPAPTATIR
proteínas/Rv3456c:VTSEANRARRVAAAQAKAKAAAMPTEESEAKPAEEGDVVGASEPDAKAPEEPPEAPEN
proteínas/Rv3691:LAKAPGDLLLVAPTSRTALTPLRLIAAASPFNSQPNCTLRANRAGSVQWGPSTYQA
proteínas/Rv3808c:YEALKN TDCQQILFMDDDIRLEPDSILRVLAMHRFAKAPMLVGGQMLNLQEPShLHIMGE
```

7. 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.

```
~/Downloads/sample
> grep -c "$AKAP" proteínas/*
proteínas/Rv0001:10
proteínas/Rv0002:8
proteínas/Rv0003:8
proteínas/Rv0004:5
proteínas/Rv0005:13
proteínas/Rv0006:15
proteínas/Rv0007:7
proteínas/Rv0008c:4
proteínas/Rv0009:5
proteínas/Rv0010c:4
proteínas/Rv0011c:3
proteínas/Rv0012:6
```

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.

```
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`

```
~/Downloads/sample
> find . -mtime +2
./proteinas
./proteinas/Rv1300
./proteinas/Rv2270
./proteinas/Rv3512
./proteinas/Rv2152c
./proteinas/Rv3349c
./proteinas/Rv2866
./proteinas/Rv0284
./proteinas/Rv3844
./proteinas/Rv0424c
./andromeda.jpg
./sequence.fasta
./SOAPdenovo2.pdf
./pan-genome-size.pdf
./Poster.pdf
~/Downloads/sample
>
```

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

```
~/Downloads/sample
> find . -mtime -1
.
./mt.fasta
./mt2.fasta
```

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

```
~/Downloads/sample
> find . -size +500k
./mt.fasta
./mt2.fasta
./Poster.pdf
```

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`

```
~/Downloads/sample
> find . -size +500k -ls
170641 1368 -rw-rw-r-- 1 igor igor 1394290 set 13 15:45 ./mt.fasta
170544 1368 -rw-rw-r-- 1 igor igor 1394290 set 13 15:46 ./mt2.fasta
131335 604 -rw-rw-r-- 1 igor igor 611453 mai 5 2014 ./Poster.pdf
```

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

```
~/Downloads/sample
> find . -name '*.pdf' -ls
149622    376 -rw-rw-r--  1 igor    igor      382965 jan  7  2014 ./SOAPdenovo2.pdf
149623    316 -rw-rw-r--  1 igor    igor      322808 fev  24  2014 ./pan-genome-size.pdf
131335    604 -rw-rw-r--  1 igor    igor      611453 mai  5  2014 ./Poster.pdf
```

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`

```
~/Downloads/sample
> find . -perm -o=x
.
./proteinas
./mtv2
```

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

```
~/Downloads/sample
> find . -perm -g=w -type d
.
./proteinas
./mtv2
```

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`

```
Terminal
~/Downloads/sample
> find ./proteinas -size +1k -type f -ls
160708    4 -rw-rw-r--  1 igor    igor      1105 mai  5  2014 ./proteinas/Rv3512
150123    4 -rw-rw-r--  1 igor    igor      1361 mai  5  2014 ./proteinas/Rv0284
168566    4 -rw-rw-r--  1 igor    igor      2575 mai  5  2014 ./proteinas/Rv3343c
160669    4 -rw-rw-r--  1 igor    igor      1413 mai  5  2014 ./proteinas/Rv3507
158953    4 -rw-rw-r--  1 igor    igor      1572 mai  5  2014 ./proteinas/Rv2932
161742    4 -rw-rw-r--  1 igor    igor      1125 mai  5  2014 ./proteinas/Rv0631c
149869    4 -rw-rw-r--  1 igor    igor      2562 mai  5  2014 ./proteinas/Rv0101
168574    4 -rw-rw-r--  1 igor    igor      3787 mai  5  2014 ./proteinas/Rv3350c
168005    4 -rw-rw-r--  1 igor    igor      1235 mai  5  2014 ./proteinas/Rv2922c
166461    4 -rw-rw-r--  1 igor    igor      1221 mai  5  2014 ./proteinas/Rv2124c
168568    4 -rw-rw-r--  1 igor    igor      1573 mai  5  2014 ./proteinas/Rv3345c
168991    4 -rw-rw-r--  1 igor    igor      1429 mai  5  2014 ./proteinas/Rv3894c
161698    4 -rw-rw-r--  1 igor    igor      1337 mai  5  2014 ./proteinas/Rv0578c
157671    4 -rw-rw-r--  1 igor    igor      1067 mai  5  2014 ./proteinas/Rv1536
164038    4 -rw-rw-r--  1 igor    igor      1361 mai  5  2014 ./proteinas/Rv1450c
165375    4 -rw-rw-r--  1 igor    igor      1493 mai  5  2014 ./proteinas/Rv1917c
161053    4 -rw-rw-r--  1 igor    igor      1091 mai  5  2014 ./proteinas/Rv3728
161491    4 -rw-rw-r--  1 igor    igor      2250 mai  5  2014 ./proteinas/Rv0304c
166400    8 -rw-rw-r--  1 igor    igor      4230 mai  5  2014 ./proteinas/Rv2048c
168136    4 -rw-rw-r--  1 igor    igor      1138 mai  5  2014 ./proteinas/Rv3080c
156503    4 -rw-rw-r--  1 igor    igor      1200 mai  5  2014 ./proteinas/Rv0667
164538    4 -rw-rw-r--  1 igor    igor      1201 mai  5  2014 ./proteinas/Rv1640c
160709    4 -rw-rw-r--  1 igor    igor      1522 mai  5  2014 ./proteinas/Rv3514
164548    4 -rw-rw-r--  1 igor    igor      1037 mai  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 {} \;

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

~/Downloads/sample
> ls
andromeda.jpg  mt.fasta  pan-genome-size.pdf  proteínas  SOAPdenovo2.pdf
maiores.fasta  mtv2      Poster.pdf           sequence.fasta
```

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 {} \;

```
~/Downloads/sample
> find ./proteinas -size +1k -type f -ls -exec grep AKAP {} \;

168788 4 -rw-rw-r-- 1 lgor lgor 1185 mai 5 2014 ./proteinas/Rv3512
150123 4 -rw-rw-r-- 1 lgor lgor 1361 mai 5 2014 ./proteinas/Rv0284
168566 4 -rw-rw-r-- 1 lgor lgor 2575 mai 5 2014 ./proteinas/Rv3343c
168669 4 -rw-rw-r-- 1 lgor lgor 1413 mai 5 2014 ./proteinas/Rv3507
159953 4 -rw-rw-r-- 1 lgor lgor 1572 mai 5 2014 ./proteinas/Rv2932
161742 4 -rw-rw-r-- 1 lgor lgor 1125 mai 5 2014 ./proteinas/Rv0631c
149869 4 -rw-rw-r-- 1 lgor lgor 2562 mai 5 2014 ./proteinas/Rv0101
168574 4 -rw-rw-r-- 1 lgor lgor 3787 mai 5 2014 ./proteinas/Rv3350c
169805 4 -rw-rw-r-- 1 lgor lgor 1235 mai 5 2014 ./proteinas/Rv2922c
166461 4 -rw-rw-r-- 1 lgor lgor 1221 mai 5 2014 ./proteinas/Rv2124c
168568 4 -rw-rw-r-- 1 lgor lgor 1573 mai 5 2014 ./proteinas/Rv3345c
168991 4 -rw-rw-r-- 1 lgor lgor 1429 mai 5 2014 ./proteinas/Rv3894c
161598 4 -rw-rw-r-- 1 lgor lgor 1337 mai 5 2014 ./proteinas/Rv0576c
157671 4 -rw-rw-r-- 1 lgor lgor 1067 mai 5 2014 ./proteinas/Rv1536
164038 4 -rw-rw-r-- 1 lgor lgor 1361 mai 5 2014 ./proteinas/Rv1450c
165375 4 -rw-rw-r-- 1 lgor lgor 1493 mai 5 2014 ./proteinas/Rv1917c
161953 4 -rw-rw-r-- 1 lgor lgor 1091 mai 5 2014 ./proteinas/Rv3720
161491 4 -rw-rw-r-- 1 lgor lgor 2250 mai 5 2014 ./proteinas/Rv0304c
166400 8 -rw-rw-r-- 1 lgor lgor 4230 mai 5 2014 ./proteinas/Rv2048c
160136 4 -rw-rw-r-- 1 lgor lgor 1130 mai 5 2014 ./proteinas/Rv2080c
156503 4 -rw-rw-r-- 1 lgor lgor 1200 mai 5 2014 ./proteinas/Rv0667
164538 4 -rw-rw-r-- 1 lgor lgor 1201 mai 5 2014 ./proteinas/Rv1640c
160709 4 -rw-rw-r-- 1 lgor lgor 1322 mai 5 2014 ./proteinas/Rv1514
164548 4 -rw-rw-r-- 1 lgor lgor 1037 mai 5 2014 ./proteinas/Rv1651c
156504 4 -rw-rw-r-- 1 lgor lgor 1346 mai 5 2014 ./proteinas/Rv0668
161591 4 -rw-rw-r-- 1 lgor lgor 1574 mai 5 2014 ./proteinas/Rv0425c
AAQLSGYLLARKVDAQAPPAHEHMSYEQKALPSPFEQAPAPSPSPAPABA
157288 4 -rw-rw-r-- 1 lgor lgor 1261 mai 5 2014 ./proteinas/Rv1161
157876 4 -rw-rw-r-- 1 lgor lgor 2170 mai 5 2014 ./proteinas/Rv1661
```

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 uma 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

Primeiro print: execução do comando. Segundo: resultado com os < ACP > após usar o cmd “cat”.

```
~/Downloads/sample
> sed -i "s/ACP/< ACP >/g" maiores.fasta

~/Downloads/sample
> cat maiores.fasta
>Rv3512
PQGADGNAAGNGDGGVGGNGGADNTTTAAAGTTGAGGAGGAGGTGGTGAAGTGTGG
QQGNGNGNGGTGGGGTGGGALAGSGGAGGKGGGDDAGKAGTGSAPGTAGTGGD
GKCGNGGIGAAAGTGPVGTGASGGTGGSGAGGTGGDGAAGGTAGAGGAGNGGKGGD
GGAGVTSSTAGNSGGAGSGGGGDDAGGAGATPGANGIAGNGGDDGDDAAGAVTSGA
TCAGDGGHGGTGAAGNGGTGAGGSGIDVGGGTGGTGGNGGATGAGGDDAGGSGNS
GGNGGIGKGGQNAAGAGGAGNGGTGANGTGGDGGNGGAAGATAGSNGCAGTGSAGCN
GGTGGRCGSGGAGGDDTGGVGGGKGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
GGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
AGCGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
TGGCAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
NAGMGNSGTGSGDGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
TANMTAAGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
DDPGGNGGTGGNGGTGGTGGAGTGSAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
NGGDDGGTGGTGGDGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGGAGG
```

```
>Rv3447c
MNSGPACATADILVAPPPELRRSEPSLLIRLLPVVMSVATVGVMTVFLGSPATRHPT
FLAFPMMLLVSVLTAVTGCRRRHVSCTHNDRVVDYLCYLSVLRTSVTGTAAAHVSLNMT
HPDPATLWTLIGGPRMWRERRPGAADFCRIRVGVGSAPLATRLVVGQLPPAQRADPVTRAA
LRCLAAHATIDAPIAIPLRVGGPIAIDGDPKVRGLLRAMICQLAVMHSPEELLIAVG
VSDNRRAHMDWLKLPWHQHPNACDALGPAPMVYSTLAEMONALAAATVLAHVVAIVDTAE
RGNGAITGVITIEVGARRDGAPPVRCAGEVTAL< ACP >DQLEPQDALVCARLLAAHRVHGS
GRTFIRGSGMAELVGIDVADFPDSTLWRNVNQHDLRVPIGVTPDGTAVQLDIKEAAEQ
GMGPHGLCVGATGSGKSELLRTIALGMMARNSPEVLNLLVDFKGGATFLDLGAPHVAA
VITNLAEAPLVARMQDALAGMSRQQLLRMAGHLVSVTAQYQRQTGAQLPCLPILFI
VVDEFSELLSQHPEFVDVFLAIGRVGRSLGMHLLASQRLDEGRRLGLETHLSYRMLCT
WSASESRNVLTQDADYQLPNTPGAGLLQTGTGELTRFOTAFVSGPLRRASPSAVHPVAP
SVRPFTHAAAPVTAGPVGGTAEVPTPTVLHAVLDRLVHGCAHQAQVLPPLDEPMLGA
LLRDAEPQAQALAVPIGIVDRPFQSRVPLTIDLGAAGNVAVVGPOTGKSTALRTLIM
ALAAATHADGRVOFYCLDFGGALAQVDLPHVCAVAGRAQPLASRLMAELASAVRFREA
FFRDHIDSVARVYQLRAKSAESFADIFLVIDGMSLROFEALLESIVALAAQGLSFG
VHVALSAARWAEIRPSLRDQTCRIELRLADPASELDRROAQRPVDRPRGLSRDGMH
MVIALPDLGVALRRSGDPVAPPILPLPARVDYDSVVARAGDELGAHLLGLEERRQGP
VAVDFGRPHLLVLGDNCGCKTAALRLCREIVRTHTAARQALLIVDFRHTLLDVIESEH
MSGVSSPAALGAKLSSLDLQARMPPADPVSAQQLRARSWMSGPDIVVVDDYDLVAVS
SGNPLMVLLEYLPHARDLGLHLVARRSGGAARALFEPVLASRLDGCRAALLMSGRPDEG
ALFSGSRMPLPPGRGILLVTGAGDEQLVQVAVSPPP
```


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/*`

```
~/Downloads/sample
> grep "AK[ARP]P" proteínas/*
proteínas/Rv0338c: CQSQC PAWNTGKPLSPKLVIMDLRDHWMAKAPYILGQKDASAGGEAGHQEHHPVESGFG
proteínas/Rv0338c: MAAGAKRPGAKKAAPTTPAAPAAPAPVKGLGIAAGAKRPGAKKT PPPAPGLAEPAAQPP
proteínas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAPAHWHAMSVEQVRKALPSPDEQAPAKAPSPYPARA
proteínas/Rv0540c: MSCLPVSVLVAKAPEPGRVKTRLAAAI GDKVAADIAAAALLDTLDAVAAAPVTARAVAL
proteínas/Rv0573c: DTGYSYPVAVSDRIVGELARLRHADTAEAHPGSNNVVGAKAKRP
proteínas/Rv0581: MDKTTVYLPDELKAAVKRAARQGVSEAVQVIREIRA AVGGAKPPPRGGLYAGSEPIARR
proteínas/Rv0586: TLAALTTTMTAAAKRPSDYRKLA DAICSGDPTGAKKAAQDLLELANTSLMAVLVQSASRQ
proteínas/Rv0706: PAKDQSAKSSRRARREASKAASKVGATAPAKKAAAKAPAKKAPASSGVKKTPAKKAPAK
proteínas/Rv0949: MSVHATDAKPPGSPADQLDGLNPQQRQAVVHEGSPLLIVAGAGSGKTAVLTRRIAYLM
proteínas/Rv0983: GSGIILSAEGLILTNNHVIAAAAKPLLGSPPPKTTVTFS DGRTPAFFTVVGADPTSDIAVV
proteínas/Rv1024: MPEAKRPESKRRSPASRPGKAGDSVRGGRATKPSAKPSTPAPHASRKTTRTPHEHIVEPI
proteínas/Rv1215c: MARNPSPALDRPWRRPGALRYALERVGVAKPPITVTDPPADVIERDVEVPTRDGTLRL
proteínas/Rv1425: KADVGNQVSSMTASLATHIEDPAKRLAAIHESTLSAKEMAKAPSAHQIMGLTETTPGGLL
proteínas/Rv1426c: ATLPTPEMRSRGRNPLR TAMARRRYVETTNNVVCYGPYGRANLADIWRRRLDPRDAKAPV
proteínas/Rv1483: MTATATEGAKPPFVSRSLVTGGNRGIGLAIQRLAADGHKVAVTHRGSGAPKGLFGVEC
proteínas/Rv1484: MTGLLDGKRILVSGIITDSSIAFHIA RVAEQGAQLVLTGFDRLLRITQRIITDRAKAPL
proteínas/Rv1530: VAALTEQTGGLADVVDVTAKAPAAFAQAIALARPAGTVVVAGTRGVSGGAPGFS PDVVV
proteínas/Rv1660: TWRSLGEIGNLSSASVLHVLRDTIAKPPPSGSPGLMIAMGPGFCSELVLLRWH
proteínas/Rv1731: TDWAKRPVIERAAVIRRYRDLV IENREFLMDLLQAEAGKARWAAQEEIVDLIANANYAR
proteínas/Rv1985c: GFTAAAAAKAPSLAWNRRDGLQDMLVRKAFRRAITRPTHFVPTTEGFTAAARAGL GWGMF
proteínas/Rv2164c: QTSPLSPFDRPAPAKNTSQAKARAKARKAKAPKLV RPTPMERLAARLTSIDL RPTLAN
proteínas/Rv2183c: EQHPLLTVIADHSLALLVIRATVDDIDRSAKPPEGPPGGGGQTASGGGENTGEGSMKS
proteínas/Rv2215: APYVTPLVRLKASENNIDLAGVTGTGVGGRIRKQDVLAAAEQKKRAKAPAPAAQAAAAPA
proteínas/Rv2448c: DFLAKAPDAVIAKIRDRQVAAQQUETERITTRLAALQ
```

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/*`

```
~/Downloads/sample
> grep "AK.P" proteínas/*
proteínas/Rv0014c: EGLSADLDVAVL KALAKNPENRYQTA AEMRADLV RVHNGEPPEAPKVL TDAERTSLLSSA
proteínas/Rv0060: ELLASTHWATREGAKEPATAAAAVRKWTKRKGRISDDRIGVALDRILMTA
proteínas/Rv0061: LLAVAHGQVAKTPSATRAIAFRHVRLMRVRWICAGNRGRKHRRCTTQYRSTQASKQLQH
proteínas/Rv0072: PRVSEGRSPSKPDEVAASSTMGRHLGDTVEVGARRLRVVGIVPNSTALAKIPNVFLTTEG
proteínas/Rv0126: ERDYMYAEYAKDPRMKANVGIRRLAPLLDNRNQIELFTALLSLPGSPVLVYGGDEIGM
proteínas/Rv0147: AVRERVIREVPAGGMMVNHLAFQVSTAKLPFGGVGASGMGAYHGRWGFEFSSHRSKSVLTK
proteínas/Rv0156: LVLQVILFVAVVFGTLNVIGGFIVTDRMLGMFKAKKPAVPAKPDREALR
proteínas/Rv0197: RVVLNEIAKDPGRSMIVIDPVVTD TAKMADFHLRVQPGCDAWCLAALAAVLVQENL CNEA
proteínas/Rv0206c: VGGTPALELDSIHGLFAKMPLLMVVILLTTTIVLMFLAFGSVVLPIKATLMSALT LGSTMG
proteínas/Rv0213c: PGTPLYQAYSDAGYL TAKWPLLQWFEFVDPEASRVYADVAVKAPDVGISFDEAEAYFLSR
proteínas/Rv0283: SSYALKDSGKTI SDTVQYYAVLPDGLQQISPVLAAILRNNNSYGLQQPPRLGADEVAKLP
proteínas/Rv0309: PNHWWSGDDNSPTFNSMQVCQKSQC PFSTADSENLIQIPQYKHSVVMGVNKAKVPGKGSF
proteínas/Rv0338c: CQSQC PAWNTGKPLSPKLVIMDLRDHWMAKAPYILGQKDASAGGEAGHQEHHPVESGFG
proteínas/Rv0338c: MAAGAKRPGAKKAAPTTPAAPAAPAPVKGLGIAAGAKRPGAKKT PPPAPGLAEPAAQPP
proteínas/Rv0353: MAKNPKDGESRTFLISVAAELAGMHAQTLRTYDRLGLVSPRRTSGGGRYSLHVDVLLRQ
proteínas/Rv0363c: SGLGVKDMVTGAVALAEFTHVIAAKYPVNVALHTDHC PKDKLSYVRPLLAISAQVRSKG
proteínas/Rv0370c: GVAETIDWVAALVALGVADLTADSSPALASLGALAKTPDDRTQIRDAVQAFTECSHA
proteínas/Rv0372c: LVASTVRGASILDSLSDAERARVHTPVGLAIGAKTPAEIAVSI AAEIATLRGGGPRG
proteínas/Rv0373c: TGFARDYIMVGEIAANRDGKILAIRSNVLADHGAFNAQAAPAKYPAGFFGVFTGSYDIEA
proteínas/Rv0375c: CTVLDAVCLAKGPSGEREIAIDDFLVGPYETALAHNEVLIEVRIPLRHNTSSAYAKVERR
proteínas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAPAHWHAMSVEQVRKALPSPDEQAPAKAPSPYPARA
proteínas/Rv0444c: AEQVLTAPDVRTVSRPLGAGTATVVF SRDRNTGLLVMMNVAPP SRGTVYQMWLLGAKGP
proteínas/Rv0447c: MTVETSQTPSAAIDSDRWPAVAKVPRGLAAASAAIANLLRRRTATHLPLRLVYSDGTAT
proteínas/Rv0462: TEQQARNEGVDVVAKFPFTANAKAHGVGDP SGFVKVLVADAKHGELLGGHLVGHDAELL
proteínas/Rv0469: LEPGMTLLDIGCGWGGGLQRAIENYDVNVIGITLSRNQFEYSKAKLAKIPTERSVQVRLQ
```

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 contrabarra: `grep "AKA\+P" proteinas/*`

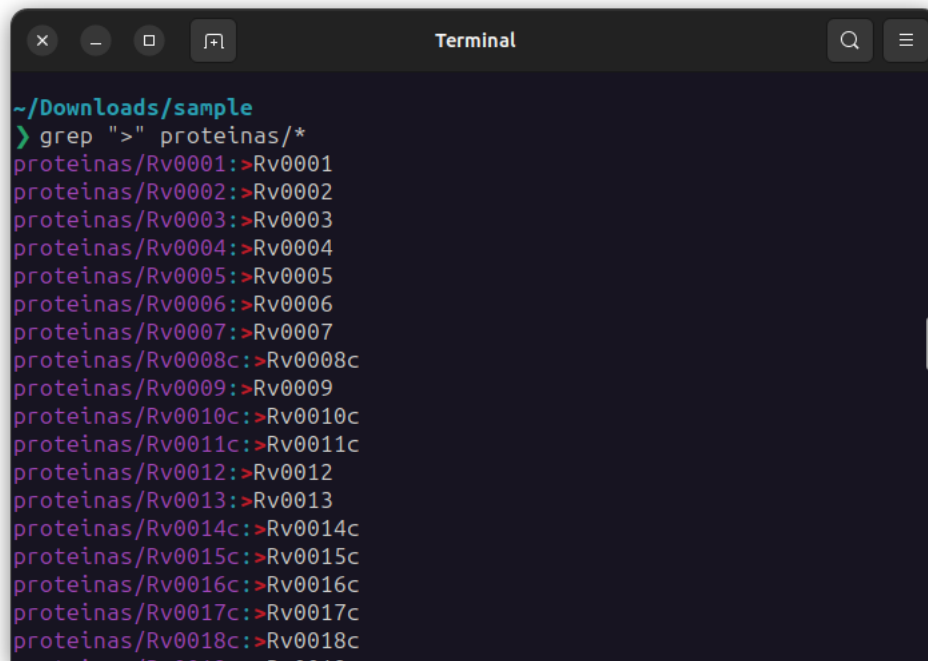
```
~/Downloads/sample
> grep "AKA\+P" proteinas/*
proteinas/Rv0338c: COSQC PAWNTGKPLSPKLVIMDLR DHWM AKAP YILGQK DASAGGEAGHQEH HHVPESGFG
proteinas/Rv0425c: AAGLLSGYLLARKVVDQAAPRPAPAHEWHAMSV EQVRKALPSPDEQAP AKAPSPSPYPARA
proteinas/Rv0540: MSCLPVSVLVV AKAP EPGRVKTRLAAAI GDKVAADIAAALLD TLD AVAAAPVTARAVAL
proteinas/Rv0706: PAKDQ RSAKSSRARRTEASKAASKVGATAPAKKAA AKAPAKKAPASSGVK KTPAKKAPAK
proteinas/Rv0710: MMAEAKTG AKAAPRVAKA AKAAPKKAAPND AEAI GAANAANVKGPKHTPRTPKPRGRKT
proteinas/Rv1425: KADVGNQVSSMTASLATHIEDPAKRLAAIH ESTLSAKEM AKAPSAHQIMGLTETTTPPGLL
proteinas/Rv1426c: ATLPT EPMRSGRNLP LRTAMARRRYVETTNNV CYGPYGRANLADIWRRRDLPRD AKAPV
proteinas/Rv1484: MTGLLDGKRILVSGIITDSSIAFH IARVAQEQQALVLTGFDRLRLIQRITDRLP AKAPL
proteinas/Rv1530: VAALTEQTGGLADVVDVT AKAPAAFAQAIALARPAGTVV VAGTRGVGSGAPGFSPDVVV
proteinas/Rv1657: MSRAK AAPVAGPEVAANRAGRQARIVAILSSAQVRSQNELAALLAAEGIEVTQATLSRDL
proteinas/Rv1985c: GFTAAAA AKAPSLAWNRRDGLQDMLVRKAFRRAITRPTHFVPTTEGFTAAARAGLGWGMF
proteinas/Rv2164c: QTSPLMSPFDRPAPAKNTSQAKARAKARK AKAPKLV RPTPMERLAARLTSIDL RPRTLAN
proteinas/Rv2215: APYVTPLVRLKASENNIDL AGVTGTGVGGRIRKQDVLAAAEQKKR AKAPAPAAQAAAAPA
proteinas/Rv2448c: DFLAKAPDAVIAKIRDQRVAQQETERITTRLAALQ
proteinas/Rv2536: VFLVGIVGVAVGRWLVDRL AKAPVRHGLAAEHERAADTDVFSAVRADDSP TGEMQVAQ
proteinas/Rv2703: ASAPQDTTSTIPK KRTRAAAKSAA AKAPSARGHATKPRAPKDAQHEAADPEDALDSVE
proteinas/Rv2780: VPGAKAPKLVSNSLVAHMKPGAVLVDIAIDQGGCFEGSRPTTYDHPTFAVHDTLFYCVAN
proteinas/Rv2973c: LGMVVDEQHRFGVEQRDQLR AKAPAGITPHLLVMTATPIRPTVALTVYGDLETSTLREL
proteinas/Rv3014c: DEGE L FALTERDLR TDLFR TKAGEL SANGKRLLVNLDK AKAPLWRV LVALSIRHVGP
proteinas/Rv3099c: LDEIAATMGFVVGKL AKAPDGSRVLLELTGPLSR SIRVSDGRARVDDFGGPAPTATIR
proteinas/Rv3144c: TESARAPEAASAPPEAVVEVPEL EVPAMGVLPTVDPKVA AKAPLSTTRVQSGAGSGIP
proteinas/Rv3269: MAIQVFLAKATTVTITGLAGVTAYEILK KAA AKAPLRQTAVSAAALGRGTRKAEAAES
proteinas/Rv3456c: VTSEANRARRVAAAQAKAKAAAMPTEESEAKPAEEGDVVGASEPD AKAPEEPPAEAPEN
proteinas/Rv3691: LAKAPGDLLLVAPT SRTRTALT PQLRTAAASPFNSQPNCTLRANRAGSVQWGPSDTYQA
proteinas/Rv3803c: MKGRSALLRALWIAALS FGLGGVAVAAEPT AKAAPYENLMVPSPSMGRDIPVAFLAGGPH
proteinas/Rv3808c: YEALKNTDCQQLFMDDDIRLEPDSILRVLAMHRF AKAPMLVGGQMLNLQEP SHLHIMGE

~/Downloads/sample
>
```

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

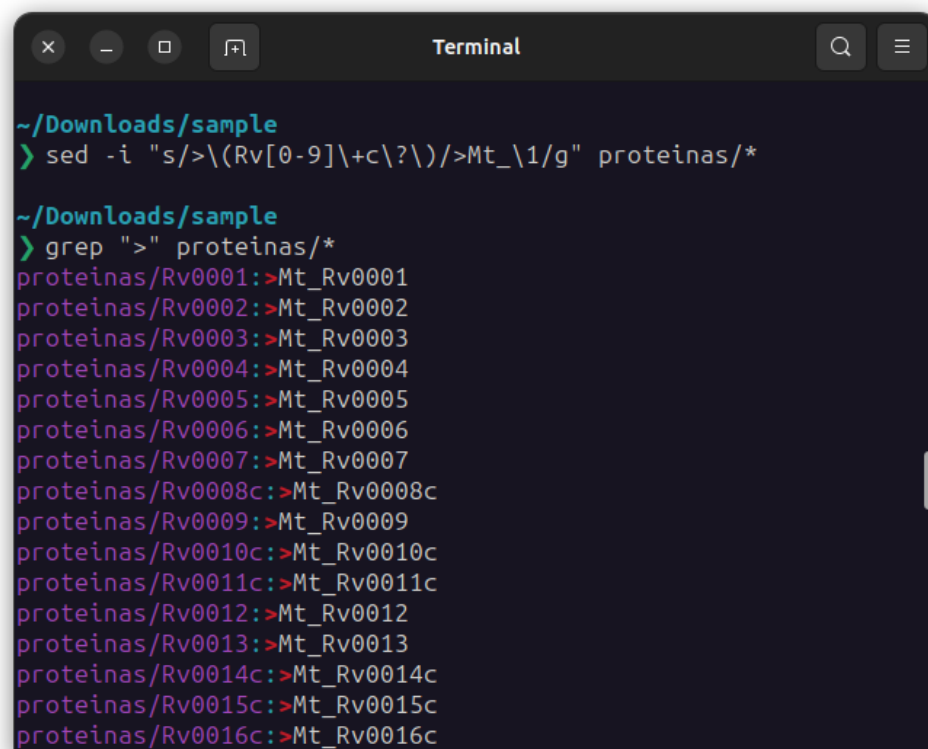
```
~/Downloads/sample
> grep "AKA*" proteinas/*
proteinas/Rv0006: LIELLDIDEIQAQAILDMQLRRLAALERQRIIDDLAKIEAEIADLEDIL AKPERQRGIVR
proteinas/Rv0033: TVGDVYTSVAVWF PET AKPAPLGKGT A
proteinas/Rv0038: ETAVYNNVLPQWAKLA AKPKTMF IGGPVKRDAALCLAVLRVGADPEGV PGLRHVAGRLVMV
proteinas/Rv0048c: TSDPG AKPDGIAPVVLTPPRQLHSLGGLTG LLEQTRKRF GDTMGYRLVIYPEYASLDRVD
proteinas/Rv0090: DLPVLRAGFKVDQMTVIHALEKALA AKPSTLALITGMLAAYAVLQAVEGVGLWLLKRWG
proteinas/Rv0095c: RITKPEAGRRSAE AKP
proteinas/Rv0102: FVSVAEPVGFFAASLAGALCLGALIHVMT AKPEPDGLIDAAAFRIHLLAERVSGLWLGL
proteinas/Rv0103c: E AKPVRAYPAAASVVGTVMDGRLVIEATAVGADTQFAAMVRLVEQAQTQKARAQRLADH
proteinas/Rv0130: GTVQATVSTTVEVEGS AKPACVAESIVRYVA
proteinas/Rv0156: LVLQVILFVAVVFGTLNVIGGFIVTDRMLGMFKAKPAVP AKPDRDEALR
proteinas/Rv0165c: MIKHDVVWVTLWPERPNKPPSPRPVGNPGPTLKLVLASHVNAPLS AKPRSQPLLRRAQ
proteinas/Rv0171: MRTL EPPNMRIGLMGIVVALLVAVGQSFTSVPM LF AKPSYYGQFTDSGGLHKGDRVRI
proteinas/Rv0189c: MPQTDEAASVSTVADIKPRS RVDTDGLEKAAARGMLRAVGMDDED F AKPQIGVASSWNE
proteinas/Rv0192: AGQDPTSFVGPPFPPTFNPNVDGAMVG V AKPIVINFAVPIADRAMAESAIHISSIPPVP
proteinas/Rv0199: ATSVSENAG AKPQTVHWNLRLDVSDVDGKLMISRLESIR
proteinas/Rv0207c: GFAVF AKPKVDESDSDVRDMLAHIDERYREGLAALVVASADGQAFRQPLEAVARS GTPVQ
proteinas/Rv0242c: VVGTP EAASTNERIAQRALEGFTRSLGKELRRGATTALVYLSPD AKPAATGLESTMRF
proteinas/Rv0269c: LLMLAEELGPPQKAQS AKPLIEIARAKTRAEMAALDIWRDRYPGAAALLRPADVLVDGM
proteinas/Rv0291: GLNATEVVRRLTATAHRA GRESSNIVGAGNLDAVAALTWQLPAEPGGGAAP AKPVADPPV
proteinas/Rv0338c: COSQC PAWNTGKPLSPKLVIMDLR DHWM AKAP YILGQK DASAGGEAGHQEH HHVPESGFG
proteinas/Rv0338c: E AKPQPEPAAPPKQTDGDPAAAPVKGLGIARGARPPGKR
proteinas/Rv0357c: QRDLCRAKPVYEELPGW WEDISGAREFDDLPAKARDYVLRLEQLAGAPVSCIGVGPGR EQ
proteinas/Rv0363c: PAD AKPFDVFHGGSGSLKSEIEEALRYGVVKMNVDTDTQYAFTRPIAGHMF TNYDGV LK
```

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



```
~/Downloads/sample
> grep ">" proteínas/*
proteínas/Rv0001:>Rv0001
proteínas/Rv0002:>Rv0002
proteínas/Rv0003:>Rv0003
proteínas/Rv0004:>Rv0004
proteínas/Rv0005:>Rv0005
proteínas/Rv0006:>Rv0006
proteínas/Rv0007:>Rv0007
proteínas/Rv0008c:>Rv0008c
proteínas/Rv0009:>Rv0009
proteínas/Rv0010c:>Rv0010c
proteínas/Rv0011c:>Rv0011c
proteínas/Rv0012:>Rv0012
proteínas/Rv0013:>Rv0013
proteínas/Rv0014c:>Rv0014c
proteínas/Rv0015c:>Rv0015c
proteínas/Rv0016c:>Rv0016c
proteínas/Rv0017c:>Rv0017c
proteínas/Rv0018c:>Rv0018c
```

25. Modifique o cabeçalho de todos os arquivos dentro da pasta proteínas 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" proteínas/*`



```
~/Downloads/sample
> sed -i "s/>\\(Rv[0-9]\\+c\\?\\)/>Mt_\\1/g" proteínas/*

~/Downloads/sample
> grep ">" proteínas/*
proteínas/Rv0001:>Mt_Rv0001
proteínas/Rv0002:>Mt_Rv0002
proteínas/Rv0003:>Mt_Rv0003
proteínas/Rv0004:>Mt_Rv0004
proteínas/Rv0005:>Mt_Rv0005
proteínas/Rv0006:>Mt_Rv0006
proteínas/Rv0007:>Mt_Rv0007
proteínas/Rv0008c:>Mt_Rv0008c
proteínas/Rv0009:>Mt_Rv0009
proteínas/Rv0010c:>Mt_Rv0010c
proteínas/Rv0011c:>Mt_Rv0011c
proteínas/Rv0012:>Mt_Rv0012
proteínas/Rv0013:>Mt_Rv0013
proteínas/Rv0014c:>Mt_Rv0014c
proteínas/Rv0015c:>Mt_Rv0015c
proteínas/Rv0016c:>Mt_Rv0016c
```