

Introduction

This cheatsheet contains 10 useful AWK one-liners for manipulation of FASTA files. It is created as part of a series to help graduate students and biologists in learning some simple programming scripts. Each oneliner is usually accompanied by additional comments which start with a hash (">"). Runnable codes is available on http://code.runnable.com/VZsPvrVQ5JkyE_ru/awk-one-liners-for-fasta-manipulation-for-shell-bash-and-bioinformatics

Author: Melissa M.L. Wong; Date created: 1 July 2015; Date last modified: 6 July 2015; Email: melissawongukm@gmail.com

FASTA format is a text-based format for representing either nucleotide sequences or peptide sequences, in which nucleotides or amino acids are represented using single-letter codes. A fasta sequence must start with an arrow (">"), followed by its name and a newline character ("\n"), and lastly its sequence which can span multiple lines.

1. To find sequences with matching name

```
awk 'BEGIN{RS=">";FS="\n"}NR>1{if ($1~/name/) print ">$0"}' file.fa
```

2. To extract sequences using a list

```
awk 'BEGIN{RS=">";FS="\n"}NR==FNR{a[$1]++;NR>FNR{if ($1 in a && $0!="") printf ">%s", $0}}' list file.fa
```

#The names in the list must start with ">" and each name is separated by a newline ("\n")

3. To join multiple lines into single line

```
awk 'BEGIN{RS=">";FS="\n"}NR>1{seq="";for (i=2;i<=NF;i++) seq=seq"$i"; print ">$1\n"seq}' file.fa
```

#Single line sequence is desirable when a sequence is long and spans many lines. Furthermore, single line sequence is much easier to be manipulated using AWK oneliners as showed in the next few examples.

4. To print specified sequence region

```
#To print the sequence starting from position 1 until 2213
awk 'BEGIN {RS=">"; FS=" \n"} NR> 1{s eq= " ";for (i=2;i <=N F;i++) seq=se q""$i; print " >$1 " \n"s -
ubs tr( seq ,1, 2213)}' file.fa

#To print sequence starting from position 399 until 704
awk 'BEGIN {RS=">"; FS=" \n"} NR> 1{s eq= " ";for (i=2;i <=N F;i++) seq=se q""$i; print " >$1 " \n"s -
ubs tr( seq ,39 9,7 04- 399 +1)}' file.fa

#To print sequence with matching name from position 399 until 704
awk 'BEGIN {RS=">"; FS=" \n"} NR> 1{s eq= " ";for (i=2;i <=N F;i++) seq=se q""$i; if ($1~/n ame/) print
" >$1 " \n"s ubs tr( seq ,39 9,7 04- 399 +1)}' file.fa

#Useful to print sequence region when given start position and stop position or length
```

5. To reformat into 100 characters per line

```
awk 'BEGIN{RS=">";FS="\n"}NR>1{seq="";for (i=2;i<=NF;i++) seq=seq""$i;a[$1]=seq;b[$1]=length(seq)}END{for
(i in a) {k=sprintf("%d", (b[i]/100)+1); printf ">%s\n",i;for (j=1;j<=int(k);j++) printf "%s\n",
substr(a[i],1+(j-1)*100,100)}}' fasta.txt
```



By [melissamlwong](#)

Published 2nd July, 2015.

Last updated 12th May, 2016.

Page 1 of 2.

Sponsored by [Readable.com](#)

Measure your website readability!

<https://readable.com>

6. To substitute nucleotide sequences

```
#To substitute small letter with capital letter
awk 'BEGIN {RS=">"; FS=" \n"} NR>1{p rinf " >%s \n", $1;for (i=2;i <=NF;i++) {gsub( /c/ ,"C", -
$i) ;gs ub( /a/ ,"A", $i) ;gs ub( /g/ ,"G", $i) ;gs ub( /t/ ,"T", $i); printf " %s \n", $i}}' file.fa
```

7. To convert DNA to RNA

```
awk 'BEGIN{RS=">";FS="\n"}NR>1{printf ">%s\n", $1;for (i=2;i<=NF;i++) {gsub(/T/,"U", $i); printf
"%s\n", $i}}' file.fa
```

8. To summarize sequence content

```
awk 'BEGIN{RS=">";FS="\n";print
"name\tA\tC\tG\tT\tN\tlength\tGC%"}NR>1{sumA=0;sumT=0;sumC=0;sumG=0;sumN=0;seq="";for (i=2;i<=NF;i++)
seq=seq""$i; k=length(seq); for (i=1;i<=k;i++) {if (substr(seq,i,1)=="T") sumT+=1; else if
(substr(seq,i,1)=="A") sumA+=1; else if (substr(seq,i,1)=="G") sumG+=1; else if (substr(seq,i,1)=="C")
sumC+=1; else if (substr(seq,i,1)=="N") sumN+=1}; print
$1"\t"sumA"\t"sumC"\t"sumG"\t"sumT"\t"sumN"\t"k"\t"(sumC+sumG)/k*100}' file.fa
#Calculate number of each nucleo tide, total length and GC content
```

9. To reverse complement nucleotide sequences

```
awk 'BEGIN{RS=">";FS="\n";a["T"]="A";a["A"]="T";a["C"]="G";a["G"]="C";a["N"]="N"}NR>1{for (i=2;i<=NF;i++)
seq=seq""$i;for(i=length(seq);i!=0;i--) {k=substr(seq,i,1);x=x a[k]}; printf ">%s\n%s", $1,x}' file.fa
#This will produce a single line sequence
```

10. To convert FASTQ to FASTA format

```
awk 'NR%4==1{print ">"substr($0,2)}NR%4==2{print $0}' file.fq
#print first and second line of every four lines. Replace the first character of the first line with " -
>".
```



By melissamlwong

Published 2nd July, 2015.

Last updated 12th May, 2016.

Page 2 of 2.

Sponsored by [Readable.com](https://readable.com)

Measure your website readability!

<https://readable.com>

cheatography.com/melissamlwong/