Introduction to high-throughput sequencing file formats – practical exercises

Daniel Vodák
(Bioinformatics Core Facility,
The Norwegian Radium Hospital)
(danielvo@ifi.uio.no)

All example files are located in directory */data/file_formats* .

Questions:

a) FASTA

File *uniprot-kinase.fasta* contains a collection of kinase sequences obtained from the UniProt database (http://www.uniprot.org/).

1) Count the number of lines in the file.
2) Count the number of header lines (i.e. the number of sequences) in the file.
3) Count the number of header lines with "Homo" in the organism/species name.
4) Count the number of header lines with "Homo sapiens" in the organism/species name.
5) Display the header lines which have "Homo" in the organism/species names, but only such that do not have "Homo sapiens" there.

b) FASTQ

File *NKTCL_31T_1M_R1.fq* contains a reduced collection of tumor read sequences obtained from The Sequence Read Archive (http://www.ncbi.nlm.nih.gov/sra).

1) Count the number of lines in the file.
2) Count the number of lines beginning with the "at" symbol ("@").
3) How many reads are there in the file?

c) SAM

File *NKTCL_31T_1M.sam* contains alignments of reads stored in files *NKTCL_31T_1M_R1.fq* and *NKTCL_31T_1M_R2.fq*.

1) Which program was used for the alignment?
2) How many header lines are there in the file?
3) How many non-header lines are there in the file?
4) What do the non-header lines represent?
5) Reads from how many templates have been used in the alignment process?

Answers:

a) FASTA

1) wc -l uniprot-kinase.fasta
2) grep "^>" uniprot-kinase.fasta | wc -l
3) grep "^>" uniprot-kinase.fasta | grep "OS=Homo" | wc -l
4) grep "^>" uniprot-kinase.fasta | grep "OS=Homo sapiens" | wc -l
5) grep "^>" uniprot-kinase.fasta | grep "OS=Homo" | grep "OS=Homo sapiens" -v

b) FASTQ

1) wc -l NKTCL_31T_1M_R1.fq
2) grep "^@" NKTCL_31T_1M_R1.fq | wc -l
3) A well-formed FASTQ file will have 4 times fewer reads than lines (1 million reads in this case).

c) SAM

1) One can find out by looking at the header line marked with tag "PG":
   grep "^@PG" NKTCL_31T_1M.sam
2) grep "^@" NKTCL_31T_1M.sam | wc -l
3) grep "^@" -v NKTCL_31T_1M.sam | wc -l
4) Alignments of individual template segments as well as unaligned segments.
5) grep "^@" -v NKTCL_31T_1M.sam | cut -f 1 | sort | uniq -c | wc -l