Exercise 1

After you have signed in to the HPC and initiated an interactive session with interactive-bio-ds, take a look around, practising the pwd, cd, ls and ls with flags. When you are happy with the concept of the working directory and navigating between directories, have a go at the tasks below.

1. Look at the man for ls, work out what the flag -lrth does to the output
2. Use the command mkdir to make a file structure which you can understand. You need to have a directory for input data, and some for the outputs of sessions 1-3. Remember the naming conventions in Linux – do not use spaces in your directory or file names
3. Use the cat command to look some of the files used in the last session, then compare how useful this is compared to looking at them with less. Remember that you exit less by pressing q.
4. Use the head command to look at top lines of the documents, and use -n flag to change the number of lines you look at. Think about how this may be useful when working with many different file types, or custom-built files with no standard format.
5. Using cp or mv, to put the files generated in the last session into the file structure you have created.
6. Use nano to create a file (with a name you define yourself, ending with the extension ‘.txt’) with the following headers in it. For each line copy the header and then complete the line. Use Ctrl + o to save the file and Ctrl + x to exit nano.

Project Title: <most recent research project title>

Project Description: <1 sentence describing your project>

Project Keywords: <at least 3 keywords related to your project>

Project Dates: <Start + end date of your project>

Exercise 2

Using the .txt file that you wrote in the first session describing your research, go the following:

1. Edit the file to add a line detailing what your undergraduate degree was. Use the header ‘Previous Research: ’

nano <file\_name>, ctrl + o, ctrl + x

1. Find the number of lines in the file

wc -l <file\_name>

1. Search for the line which describes your Project

grep ‘Description’ <file\_name>

1. Count the number of lines containing the word ‘the’ in the description, and the term ‘Project’.

grep -c ‘the’ / ‘PhD’ <file\_name>

1. Without using a text editor, change the colon at the ends of your headers to an equals sign

sed 's/: /=/g' <file\_name>

sed -i 's/: /=/g' <file\_name>

Exercise 3

Using the annotation file for Human chromosome 13, use the tools introduced today to:

First download the .gff3 annotation for the Human chromosome 13. Use the two lines below to do this, but make sure you are in the correct place in your file structure to do it.

wget <https://ftp.ensembl.org/pub/current_gff3/homo_sapiens/Homo_sapiens.GRCh38.113.chromosome.13.gff3.gz>

gunzip Homo\_sapiens.GRCh38.113.chromosome.13.gff3.gz

Now use your new Linux skillz to do the following:

1. Find out how many genes are found on chromosome 13

cut -f3 Homo\_sapiens.GRCh38.94.chromosome.13.gff3 | grep -c -w 'gene'

1. Create a new file containing only the full lines describing genes

cut -f3 Homo\_sapiens.GRCh38.94.chromosome.13.gff3 | grep -w 'gene' > <new>

1. Find out how many genes have a gene space > 10,000nt

awk '$3 == "gene" {print $5 - $4}' Homo\_sapiens.GRCh38.94.chromosome.13.gff3 | grep -c 1

1. Find what gene is associated with the Ensembl i.d. ENSG00000139618

grep ‘ENSG00000139618’ Homo\_sapiens.GRCh38.94.chromosome.13.gff3

1. For the gene which is the answer to C, work out the length of the transcript (exons only) for one of the three isoforms (different transcripts)

grep ENST00000470094 Homo\_sapiens.GRCh38.94.chromosome.13.gff3 | awk '$3 == "exon"' | awk '{print $5 - $4}' | paste -sd+ - | bc

grep ENST00000470094 Homo\_sapiens.GRCh38.94.chromosome.13.gff3 | awk '$3 == "exon" {print $5 - $4}' | awk '{s+=$1} END {print s}'

Here the paste and bc are outside the scope of the course and will only be accessible by google - either encourage them to google (‘linux find the sum’ or ‘bash find the sum’) or if they are struggling just get them to use the desktop calc…