Final task: NGS data analysis/pipelining module

Background

You are a bioinformatician in a workgroup that works with Hepatitis delta, a small human-pathogenic virus. One of your collegues analyses samples from molecular surveillance of Hepatitis delta and thus needs to compare genomes of viruses from those samples to a reference genome.

Fortunately, the NGS lab at your workplace is able to provide the complete (short) genomes from the sequencing. However, custom analysis is not part of the service.

Required analysis

For the comparison, your colleague requires the following:

- An alignment of all genomes from a sequencing run to a reference genome
- Cleanup of the alignment (removing positions from the alignment that have a low quality)
- Simple visualization of the alignment

Fortunately, you have already been able to go through the analysis manually with you colleague and you already agreed on which tools and which parameters to use. Now, you just need to automate the task in order to be able to run it quickly and reproducibly whenever new sequencing results come in.

Pipeline development task

Please develop a nextflow pipeline that:

- Takes an NCBI GenBank accession as commandline parameter (`--accession`) and downloads the FASTA file for that accession. The default accession (if none is given on the commandline) should be M21012
- Takes a list of FASTA files containing genomes. You can use the FASTA files in https://gitlab.com/dabrowskiw/cq-examples/-/tree/master/data/hepatitis as input
- Combines all of those single FASTA files into one FASTA file - this is necessary for the alignment tool used in the next step
- Runs the mafft aligner on the combined FASTA file
- Cleans up the resulting aligned FASTA file using trimal. The trimal operation needed for this cleanup (there are different options, your colleague and you looked at the outputs all of these generate together and your colleague decided that that's the one they want) is `-automated1`.
- Publishes both the cleaned up alignment and the report html file generated by trimal into an output directory

You can publish more files, but the two files that your colleague is interested in are just the resulting FASTA file and the report HTML file (that includes

a visualization of the alignment) from trimal. You can download an example visualization (the output generated from the accession M21012 and the example data linked above) from https://gitlab.com/dabrowskiw/cq-examples/-/blob/master/data/alignment_trimmed.html.

Please:

- Implement the whole pipeline in nextflow
- Use singularity containers for running mafft and trimal
- Preferably submit the task through github or gitlab:
  - Create a new repository
  - Commit and push your code (everything necessary to run the pipeline - don't forget nextflow.config!)
  - Either make the repository public and send me a link to it, or make it private and invite me (@dabrowskiw) to it
- Alternatively, mail me a zip file with all files necessary to run the pipeline (again, don't forget the nextflow.config!) - but without any unnecessary files that you wouldn't put into a git repository. Specifically don't include the work, cache, output etc. folders in the zip file! If your zip file size is more than 10kb, you're most likely doing somethign wrong!

Tips

Downloading data

Downloading a FASTA file for an accession from GenBank can be done using the following command:

```
wget "https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?
db=nuccore&id=M21012&rettype=fasta&retmode=text" -O M21012.fasta
```

In this example, the accession M21012 is used - please adapt this to your needs, and please note that in this command, the accession is given twice: Once in the URL (defining which accession to download), and once as output filename with the parameter `-O`.

Please also note that the whole command needs to be in one line, the line break is only there to make it fit on the page.

Combining files

You can combine multiple files on the commandline using cat. For instance, to combine "file1.txt" and "file2.txt" into "both.txt", you could run:

```
cat file1.txt file2.txt > both.txt
```

or, using wildcards (assuming "file1.txt" and "file2.txt" are the only text files in the directory):

```
cat *.txt > both.txt
```