BS7105 Mini Project: Having a BLAST

Briefing:

The purpose of the BS7105 mini-project is to demonstrate the skills you have acquired so far and write a Python programme that sets up local BLAST databases, runs a few BLAST searches, and provides some analysis of the BLAST results. Your programme should have **four functionalities**. Each functionality can be implemented at a different level of complexity. The level of complexity is for you to decide. Don't be overly ambitious, but use the opportunity to showcase your programming skills.

1) **Get FASTA data**
   This **functionality** should download, save and unpack protein FASTA files for three species from the latest version of wormbase: https://parasite.wormbase.org/ftp.html .

   For example, the protein FASTA file for the species *Brugia malayi* is available from the EBI-FTP server in the following directory:
   https://ftp.ebi.ac.uk/pub/databases/wormbase/parasite/releases/WBPS17/species/brugia_malayi/PRJNA10729/

   In this directory the relevant file name is:
   brugia_malayi.PRJNA10729.WBPS17.protein.fa.gz

2) **Prepare BLAST searches**
   Now you have the data and need to prepare three BLAST searchable databases. You can assume a working BLAST installation. For the second **functionality** use the instructions from the respective BS7102 practical which covered the required BLAST commands for formatting BLAST searchable databases (a bit of debugging needed!). Test and fix them, and implement them in your Python mini-project.

3) **Run BLAST searches**
   This **functionality** should run all against all BLAST searches. This should give you six output files (1 vs 2, 1 vs 3, 2 vs 1, 2 vs 3, 3 vs 1, 3 vs 2). To keep your outputs manageable, collect only one hit and one alignment. Store BLAST outputs in table format. Check BLAST options to achieve this.

4) **Analyse BLAST results**
   Now it is time to summarize the data and analyse the BLAST outputs. This **functionality** is really up to you and your opportunity to shine. I am happy to discuss your ideas with you.

Beyond the code, prepare a **USERGUIDE** that explains the functionalities, how to use the program, etc. I will read the USERGUIDE before running your programme. Make sure I receive all relevant information I need for running your programme.

Your submission will be assessed based on the functionalities you provide, the demonstrated level of your coding skills, quality of your code and the clarity of your USERGUIDE. For your USERGUIDE don't write a long-winded essay. No one wants to read dozens of pages in a USERGUIDE, be clear and concise.

In your submission, combine your script MINIPROJECT_yourstudentnumber.py and the USERGUIDE_ yourstudentnumber.pdf to a tar.gz file and submit it as attachment by the date given below.