

Final assessment DSLSR 2021-2022

Deadline: 26 November before 23:59 hours

In this assessment you are going to recreate an online bioinformatics tool that collects signature parts of sequences from GenBank files and displays a summary of them.

Required components:

- Copy the contents of the mRNA GenBank file to the online GenBank Feature Extractor at http://www.bioinformatics.org/sms2/genbank_feat.html Upload this information using the Submit button and copy the contents of the page that then appears to a text file. (Hint: copy only the content in monospaced font; you do not need to copy the title "GenBank Feature Extractor results" with it.)
- Examine the output carefully and compare it to the input you had entered. Analyze how the output was obtained from the input. (Hint: the DEFINITION, FEATURES, and ORIGIN sections are important.)
- Write a program that takes one commandline argument, a user-specified random mRNA GenBank file. The program should output a text file containing all the features in exactly the same way that the GenBank Feature Extractor does. If the input consists of a file called *mrna.gb*, then the program should write the output away in a file called *mrna_features.txt*. (Hint: first make a step-by-step plan on paper of how you are going to do this.)
- Check how the output changes if you change the output format option from "**separated**" to "**uppercased**" in the online GenBank Feature Extractor. (Hint: look not only at the capitalization, but also at the file size of the produced file.) Provide your own program with an option that allows the user to choose either the "separated" or "uppercased" format, and make sure the output matches that.
- Using the online GenBank Feature Extractor, also convert the contents of a GenBank file containing DNA into a text file containing features. If necessary, modify your own program so that it can also reproduce this output.
- Use the protein in the GenPept format. Try to convert this one too with the online GenBank Feature Extractor. Check that this (usually) does not go quite right, but make sure that your own program can do this conversion.
- Create at least the following classes: *GenbankParser* and *Feature*. The *GenbankParser* should read and process the Genbank file. The DEFINITION and ORIGIN should be stored as attributes inside objects of this type. A *GenbankParser* object should have a list with *Feature* objects extracted from the Genbank file.
- Determine where errors in your code may appear and handle these correctly.

Hand in your main program and all other modules, files needed to run your solution of this assessment

Rename your main program to `finalassignment_yourname.py`

Make sure that your code:

- ✓ is submitted on time;
- ✓ is executable and deals with any (possible) errors
- ✓ performs the required parts of the assignment correctly and completely;
- ✓ is preferably implemented as efficiently as possible;
- ✓ uses a neat and readable programming style;