P3: Python programming

GenBank hosts its own file format for storing genome data. In Exercise P1 you have written your own GenBank parser, and applied it to a given file with GenBank records. In this exercise you will use some functionality from the BioPython package to retrieve the records from GenBank in GenBank format. Specifications below. Login to bioinfm15 (10.73.216.102), where the BioPython package is installed.

Input: a file containing multiple accession numbers. Location:
http://www.bioinformatics.nl/courses/BIF-30806/docs/p3input.txt
Tasks:

- Retrieve the sequences from GenBank **in GenBank format** using the BioPython Entrez module
- Parse the GenBank file, using your GenBank Parser written in exercise P1. Don't copy the code, but import it from your P1 script (this requires your P1 script to be in the same directory as your P3 script).
- Calculate the lengths of the retrieved sequences

Output (2 files):

A. Print a tab-delimited table of accession number, organism name, and sequence length
B. Print the label and sequence of the shortest sequence in FASTA format

Hints:

- Use the Entrez.efetch() function to retrieve the sequences in GenBank format (database "nucleotides")
- As alternative for your own parser you can also experiment with the Bio.SeqIO.parse() function
- Search field descriptions for sequence database: http://www.ncbi.nlm.nih.gov/books/NBK49540/

Create a python script, containing your **name** and **student number**, that performs the described task. **Turn in your python script on BlackBoard (under P3)**.

Example

```
>>>from Bio import Entrez
>>>Entrez.email = "your_name@your_mail_server.com"
>>>handle = Entrez.efetch(db="nucleotide",
id=["FJ817486, JX069768, JX469983"], rettype="fasta")
>>>records = handle.read()
>>>print records
```