**BINF6410 Fall 2019**

**Assignment 2:**

Submit this assignment to the Courselink Dropbox that will be created. Please name the file with your username and the assignment number, for example my file would be named lewis\_a2.py

Write a Python3 program that does the following:

1. Reads in a provided FASTA file containing a nucleotide sequence. Please do not hard-code the name of the file. The name should be presented to the program as a unix command line argument.
2. Reads in a provided text file containing a series of restriction enzymes. The file contains one enzyme per line, listing first the enzyme name, followed by a semicolon (;) and then the sequence recognized by the enzyme, with a ^ character indicating the cleavage site.

*E.g: EcoRI;G^AATTC.* Like the FASTA file, the filename should be a command line argument.

1. Uses each of the restriction enzymes to find where it cuts the nucleotide sequence.
2. Prints out a report with the following information (See the end for the formatting of the report).:
   1. the names of the files used
   2. the header information from the FASTA file
   3. the length of the nucleotide sequence
   4. for each enzyme:
      1. the name of the enzyme and its sequence
      2. the number of cutting sites found (this could be zero)
      3. if cutting sites are found:
         1. how many cutting sites and how many fragments
         2. the sequences of all of the fragments that would result if the enzyme were applied. Display these in lines of 60 bases, with each line separated into groups of 10 bases. Display the sequence position the start of each line, and keep a running count between fragments
      4. if no cutting sites are found, print a clear understandable message informing the user of this.

Your program should work with any restriction enzyme and any properly formatted FASTA file. You can assume that the FASTA file will contain only one sequence. A small test-file will be provided to you along with a list of enzymes.

Finally, you may import code from a module to process the sequence from the input FASTA file. Otherwise, the project should be done with code from commands that you learned or will learn in lecture. Of course, you are welcome to research these commands and their use.

Your output should look something like this:

Restriction enzyme analysis of sequence from file testseq.fas.

Cutting with enzymes found in file enzymes.txt.

---------------------------------------------------------------

Sequence name: TESTSEQ

Sequence is 107 bases long.

---------------------------------------------------------------

There are 2 cutting sites for EcoRI, cutting at G^AATTC

There are 3 fragments:

length: 27

1 ATTATAAAAT TAAAATTATA TCCAATG

length: 23

28 AATTCAATTA AATTAAATTA AAG

length: 57

51 AATTCAATAA TATACCCCGG GGGGATCCAA TTAAAAGCTA AAAAAAAAAA AAAAAAA

---------------------------------------------------------------

There are 1 cutting sites for BamHI, cutting at G^GATCC

There are 2 fragments:

length: 73

1 ATTATAAAAT TAAAATTATA TCCAATGAAT TCAATTAAAT TAAATTAAAG AATTCAATAA

61 TATACCCCGG GGG

length: 34

74 GATCCAATTA AAAGCTAAAA AAAAAAAAAA AAAA

---------------------------------------------------------------

There are no sites for PstI.

---------------------------------------------------------------

There are 1 cutting sites for AluI, cutting at AG^CT

There are 2 fragments:

length: 87

1 ATTATAAAAT TAAAATTATA TCCAATGAAT TCAATTAAAT TAAATTAAAG AATTCAATAA

61 TATACCCCGG GGGGATCCAA TTAAAAG

length: 20

88 CTAAAAAAAA AAAAAAAAAA

Your work will be evaluated on four axes:

1. **Scope**: The extent to which your code implements the features outlined in the specification.
2. **Correctness**: The extent to which your code is consistent with the specification and is free of bugs.
3. **Design**: The extent to which your code is well written (i.e. clear, efficient, elegant, logical).
4. **Style**: The extent to which your code is readable (i.e. commented, containing aptly named variables, etc).