Homework 3

N.B.

Make sure your code is well documented with comments. Clearly mark different sections of your code corresponding to the questions.

Any bonus point is not required, it is intended to supplement your python knowledge.

Remember to document your code! For example:

```
def myfunction(arg):
    """
    This is a docstring. Surrounded by triple double-quotes. Description of
the whole function goes here.
    """
    # This is a comment, provides concise description of following code.
    pass
```

Your best resource for help is the official python documentation.

Questions

A. Sequence Tools

Write a Python module named `seq_tools_netid.py` that contains the following functions:

1. `load_fasta(filename)`
   Create a function named `load_fasta` that takes a filename (str) as input and returns the sequence(s) (`str`) found in that file assuming it is a FASTA file containing one or more sequences (using the FASTA file format).
2. `get_reverse_complement(sequence)`
   Create a function named `get_reverse complement` that takes a sequence (`str`) as input and returns the reverse of its complementary sequence (`str`). It is okay to reuse a function from previous exercises.
3. `get_transcript(sequence)`
   Create a function named `get_transcript` that takes a sequence (`str`) as input and returns the RNA transcript produced from this sequence assuming the input is the coding sense strand (Is this the same sequence as the RNA transcript produced? You should know). It is okay to reuse a function from previous exercises.
4. `get_translation(sequence, reverse_complement_flag)`
   Create a function named `get_translation` that takes a sequence (`str`) as input and returns the translated primary sequence produced from this sequence (i.e. a protein sequence) assuming it represents a coding sequence. This function should be able to properly translate a sequence representing DNA OR RNA. This function should have a second argument (boolean) which defaults to `False`. If this second argument is `True`, the function should return the translation of the reverse complement of the input sequence. Use asterisk (`*`) for the stop codon.

5. Execution and test cases:

You should also include a function that shows some test cases for each function *e.g.*Provide different sequences (some right, some wrong) to input into your functions to test the output

- `print(get_transcript('ATATDSTAV'))` should give some error messages
- `print(get_transcript('ATATTGTAC'))` should give the correct output

B. Sequence Transcription and Translation

Write a Python script named `transcribe_translate_netid.py` which should import all the functions from `seq_tools_netid.py` (both should be in the same directory). It should take a filename as input, load sequence data from this file, and print the translation of the sequence(s), each in a separate line.
If executed, this file should take 1 input argument (the filename to analyze). You should use the functions imported from `seq_tools_netid.py` to perform these tasks.

Test Your Code

Test your script by running it on the following fasta files (attached with this homework) and ensure that it runs without error and outputs **only** the translation of the sequence(s). You can also include a check for empty fasta files (useful when fasta files are empty).

Provided test files:

- Single sequence in fasta file: `one_nt_seq.fa`
- Multiple sequences in fasta file: `multiple_nt_seq.fa`

Submission

Submit the following python files to NYU classes:

1. `seq_tools_netid.py`
2. `transcribe_translate_netid.py`
3. Export your code and output as html file and submit the .html file with the results