Homework 4

N.B.

Make sure your code is well documented with comments. Clearly mark different sections of your code corresponding to the questions.
The 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 (10 Points)

Write a Python script named `regex_netid.py` that contains the following functions. You should have other functions, *e.g.* a separate function to load FASTA files, etc. You can use functions from your previous assignments or biopython. Make sure your script runs without errors. I will be testing your script using a different sequence, but please test your functions and script using the test case provided.

A. Sequence Statistics

Write a function named `seq_statistics` that takes an input FASTA file name that contains one DNA sequence, then displays the distribution of nucleotides of the **DraII** restriction enzyme recognition sequences found within the input sequence (in the 5' to 3' direction only).

Your function should use a regular expression to define the restriction enzyme recognition sequence.

For DraII the recognition sequence is 5' `RGGNCCY in the 5' to 3' direction, using the IUPAC ambiguity code.`

Use the `draII.fa` file to test `seq_statistics` function. The output can be displayed however you like but **MUST** clearly display the distribution of characters as a position weight matrix. For example, if the found sites for some example restriction enzyme were:

GTCGAC

GTCAAC

GTCAAC

GTTGAC

GTTAAC

GTCGAC

GTTGAC

GTTAAC

GTTGAC

GTCGAC

GTCAAC

GTCAAC

GTTGAC

The output could look like:

```
A 0.0 0.0 0.0 0.46 1.0 0.0
G 1.0 0.0 0.0 0.54 0.0 0.0
C 0.0 0.0 0.54 0.0 0.0 1.0
T 0.0 1.0 0.46 0.0 0.0 0.0
```

B. Restriction Site Scanning

Write another function named `restriction_site_scan` that finds and replaces **StyI** restriction enzyme recognition sequences with `NNNNNN` for both the input sequence and the reverse complement of that sequence and displays the indices of the restriction sites found in both sequences. The recognition sequence for StyI is `5' CCWWGG.`

This function can be structured however you like but must successfully accept an input string giving the filename of a FASTA file containing **one** sequence to scan. The function should use a regular expression.

The output should display the input sequence and the reverse complement with restriction sites replaced with Ns, and the indices of where the site was found and the direction it was found on, either on the original input (i.e. 5' to 3' direction) or the reverse complement. Use `styI.fa` to test the `restriction_site_scan` function. Your script can use any functions or code used in previous assignments or outlined in class or biopython. Make sure your script runs without error on the command-line

(i.e. python regex_netid.py).

Submission

Submit the following files to NYU classes:

1. regex_netid.py

2. The output as an html or txt file (it can be the output of JupyterLab if you use that).

Bonus Point (1 Point, Optional)

Since the `str.format` has been introduced in class (example below), the function can actually format strings in many useful ways. All of which is detailed in the Format String Syntax section of the python documentation. Use an example of the Format String Syntax in your code to format your output (reference to positional arguments don't count).

```
a = "big"
print("This is a {} book.".format(a))
```

Note: This is **not** a separate section! You have to use it in one of the other homework questions if you want the point.