CSCE 110: Programming I

Lab #10 (100 points)

Due: Sunday, November 6th by 11:59pm

1 Please make sure you understand the following.

For this assignment, you are only allowed to use what we have discussed during the 10 weeks of class. Please make sure to name your files correctly and incorporate user-defined functions. Do not use the `global` keyword in your programs. Otherwise, your programs will be penalized as stated in the grading rubric.

Please label your Python programs `q<num>.py`, where `<num>` is the question number. Take your time and make sure you understand everything in this lab before getting started. Also, make sure your programs match the output EXACTLY as given for each question.

2 Lab Attendance This Week

Lab attendance is optional for Monday and Tuesday labs. Attendance is required for Wednesday and Thursday labs. You can use the optional lab days this week to prepare for Exam #2 or work on the lab assignment.

3 Lab Question #1: Analyzing DNA

Write a Python program (called `q1.py`) that analyzes DNA sequences (or strings). A DNA string consists of the alphabet **A** (adenine), **C** (cytosine), **G** (guanine), and **T** (thymine).

Your program will ask the user for the name of a DNA sequence file in FASTA format, which is a text-based format for representing DNA sequences. Then, it will print a report on the entire set of sequences consisting of:

- the number of sequences in the input file,
- the total length of all sequences,
- the maximum sequence length,
- the minimum sequence length, and

- the average length of the sequences.

For each sequence, print:

- the header line,
- the length of the sequence, and
- the number of nucleotides A, C, G, and T in the sequence.

FASTA format. A sequence in FASTA format begins with a single-line description (starting with a '>'), followed by lines of sequence data. Here is the contents of `test1.fsa`, which describes two sequences (called `sequence 1` and `sequence 2` in the FASTA file).

```
1 >sequence 1
2 ATTGGGTGCGCGTGCN
3 CCTTCC
4 >sequence 2
5 aaaaaatcatactacatgtagggtaca
```

Line 1 represents the header information since it begins with a '>'. Then, the DNA string for `sequence 1` is across two lines (lines 2–3). The next sequence in the file (called `sequence 2`) is on line 4, and its DNA string is on a single line (line 5).

Programming tips. Remember, in the FASTA format, the header line ('>') will be a single-line description, but the sequence data (lines not starting with a '>') can span multiple lines. When reading the sequence data, you will want to convert every character to uppercase. To do this, use the `upper` method for strings (see below).

```
1 s = 'acgTaAt'
2 print(s)          # prints acgTaAt
3 print(s.upper())  # prints ACGTAAT
```

When designing your program, consider dividing your solution into three parts: reading the file, analyzing the data, and printing the report. When analyzing the data, focus on getting the data organized into a list(s). Once the data is organized, then write the code to print the report from the information collected in your list(s).

Example #1. At the prompt, the user enters `test1.fsa` (line 1). Next, the total, maximum, minimum, and average lengths of the two sequences are reported (lines 3–8). Finally, the length and A, C, G, and T composition for each sequence is printed (lines 10–22).

```
1 Enter a filename: test1.fsa
2
3 Report for file test1.fsa
4 Number of sequences: 2
5 Total sequence length: 49
6 Maximum sequence length: 27
7 Minimum sequence length: 22
8 Average sequence length: 24.5
```

```
9
10 >sequence 1
11 Length: 22
12 A: 1
13 C: 7
14 G: 7
15 T: 6
16
17 >sequence 2
18 Length: 27
19 A: 13
20 C: 4
21 G: 4
22 T: 6
```

Example #2. At the prompt, the user enters `test2.fsa`. The results of the analysis of this file is then reported (lines 3–43).

```
1  Enter a filename: test2.fsa
2
3  Report for file test2.fsa
4  Number of sequences: 5
5  Total sequence length: 10110
6  Maximum sequence length: 4116
7  Minimum sequence length: 313
8  Average sequence length: 2022.0
9
10 >AB037784 Homo sapiens mRNA for KIAA1363 protein
11 Length: 4116
12 A: 1175
13 C: 828
14 G: 835
15 T: 1278
16
17 >AF141315 Homo sapiens alpha-1,4-N-acetylglucosaminyltransferase mRNA
18 Length: 1292
19 A: 350
20 C: 307
21 G: 306
22 T: 329
23
24 >AJ289841 Homo sapiens partial ADRACALA gene for adracalin
25 Length: 313
26 A: 60
27 C: 84
28 G: 117
29 T: 52
```

```
30
31 >AK022451 Homo sapiens cDNA FLJ12389 fis, clone MAMMA1002671
32 Length: 3253
33 A: 706
34 C: 864
35 G: 927
36 T: 756
37
38 >AK057908 Homo sapiens cDNA FLJ25179 fis, clone CBR09204
39 Length: 1136
40 A: 319
41 C: 272
42 G: 248
43 T: 297
```