Python exercises 2

v1.

27 October 2021

2.1. Refer back to exercise 1.4, where we printed a DNA string in blocks, with a space between each block. Now further develop your code so that it displays a DNA string in the style used in GenBank records. So given the DNA sequence:

GCTGAGACTTCCTGGACGGGGGACAGGCTGTGGGGTTTCTCAGATAACTGGGCCCCTGCGCTCAGGAGGC
CTTCACCCTCTGCTCTGGGTAAAGTTCATTGGAACAGAAAGAAATGGATTTATCTGCTCTTCGCGTTGAA
GAAGTACAAAATGTCATTAATGCTATGCAGAAAATCTTAGAGTGTCCCATCTGTCTGGAGTTGATCAAGG
AACCTGTCTCCACAAAGTGTGACCACATATTTTGCAAATTTTGCATGCTGAAACTTCTCAACCAGAAGAA
AGGGCCTTCACAGTGTCCTTTATGTAAGAATGATATAACCAAAAGGAGCCTACAAGAAAGTACGAGATTT
AGTCAACTTGTTGAAGAGCTATTGAAAATCATTTGTGCTTTTCAGCTTGACACAGGTTTGGAGTATGCAA
ACAGCTATAATTTTGCAAAAAAGGAAAATAACTCTCCTGAACATCTAAAAGATGAAGTTTCTATCATCCA
AAGTATGGGCTACAGAAACCGTGCCAAAAGACTTCTACAGAGTGAACCCGAAAATCCTTCCTTGCAGGAA
ACCAGTCTCAGTGTCCAACTCTCTAACCTTGGAACTGTGAGAACTCTGAGGACAAAGCAGCGGATACAAC
CTCAAAAGACGTCTGTCTACATTGAATTGGGATCTGATTCTTCTGAAGATACCGTTAATAAGGCAACTTA
TTGCAGTGTGGGAGATCAAG

The output should look as close as possible to:

```
   1 gctgagactt cctggacggg ggacaggctg tggggtttct cagataactg ggcccctgcg
  61 ctcaggaggc cttcaccctc tgctctgggt aaagttcatt ggaacagaaa gaaatggatt
 121 tatctgctct tcgcgttgaa gaagtacaaa atgtcattaa tgctatgcag aaaatcttag
 181 agtgtcccat ctgtctggag ttgatcaagg aacctgtctc cacaaagtgt gaccacatat
 241 tttgcaaatt ttgcatgctg aaacttctca accagaagaa agggccttca cagtgtcctt
 301 tatgtaagaa tgatataacc aaaaggagcc tacaagaaag tacgagattt agtcaacttg
 361 ttgaagagct attgaaaatc atttgtgctt ttcagcttga cacaggtttg gagtatgcaa
 421 acagctataa ttttgcaaaa aaggaaaata actctcctga acatctaaaa gatgaagttt
 481 ctatcatcca aagtatgggc tacagaaacc gtgccaaaag acttctacag agtgaacccg
 541 aaaatccttc cttgcaggaa accagtctca gtgtccaact ctctaacctt ggaactgtga
 601 gaactctgag gacaaagcag cggatacaac ctcaaaagac gtctgtctac attgaattgg
 661 gatctgattc ttctgaagat accgttaata aggcaactta ttgcagtgtg ggagatcaag
```

Hint: it's a good idea to make your code into a function which has parameters for the block size and number of blocks per row, as well as the string to print. Also, see if you can ensure that the bases are always in lower case when printed regardless of the input.

2.2. Write a function to translate a DNA sequence into an amino acid sequence (don't use imported modules to do this for now). You can find the standard genetic code table here https://www.genscript.com/tools/codon-table and elsewhere. Hint: one way is to create a dictionary to hold a translation table. Use single letter amino-acid codes, and assume coding starts from the first base only.

So given a sequence:

```
aggagtaagcccttgcaactggaaatacacccattg
```

The output should look like:

```
RSKPLQLEIHPL
```

For bonus points, deal with any errors in the DNA string, e.g. incomplete codons at the end of the sequence, gaps or incorrect bases in the sequence. For now don't consider ambiguity base codes like N.

Challenge: now translate the following sequence:

ATGGATTTATCTGCTCTTCGCGTTGAAGAAGTACAAAATGTCATTAATGCTATGCAGAAAATCTTAGAGTGTCC
CATCTGTCTGGAGTTGATCAAGGAACCTGTCTCCACAAAGTGTGACCACATATTTTGCAAATTTTGCATGCTGA
AACTTCTCAACCAGAAGAAAGGGCCTTCACAGTGTCCTTTATGTAAGAATGATATAACCAAA

2.3. Write a function which generates the reverse complement of a sequence. Bonus points for dealing with gaps or incorrect base letters. So given a sequence:

```
aggagtaagcccttgcaactggaaatacacccattg
```

the output should look like:

```
caatgggtgtatttccagttgcaagggcttactcct
```

Challenge: output the reverse complement of:

```
GCTGAGACTTCCTGGACGGGGGACAGGCTGTGGGGTTTCTCAGATAACTGGGCCCCTGCGCTCAGGAG
GCCTTCACCCTCTGCTCTGGGTAAAGTTCATTGGAACAGAAAGAAATGGATTTATCTGCTCTTCGCGT
TGAAGAAGTACAAAATGTCATTAATGCTATGCAGAAAATCTTAGAGTGTCCCATCTGTCTGGAGTTGA
TCAAGGAACCTGTCTCCACAAAGTGTGACCACATATTTTGCAAATTTTGCATGCTGAAACTTCTCAAC
CAGAAGAAAGGGCCTTCACAGTGTCCTTTATGTAAGAATGATATAACCAAAAGGAGCCTACAAGAAAG
TACGAGATTTGAT
```

2.4. Combine translation and reverse complement functions to generate a six frame translation of a DNA sequence. This means you should translate three forward reading frames starting at the first, second and third base of the first codon of the forward sequence, and three reverse reading frames starting at the first, second and third base of the first codon of the reverse complement of the sequence. So given a sequence:

```
aggagtaagcccttgcaactggaaatacacccattg
```

The output should be something like:

```
Forward
1 RSKPLQLEIHPL
2 GVSPCNWKYTH
3 E*ALATGNTPI
Reverse
1 QWVYFQLQGLTP
2 NGCISSCKGLL
3 MGVFPVARAYS
```

Challenge: print the six-frame translation of:

GCTGAGACTTCCTGGACGGGGGACAGGCTGTGGGGTTTCTCAGATAACTGGGCCCCTGCGCTCAGGAGGCCT
TCACCC

2.5. Count single, di-nucleotide and tri-nucleotides in a sequence. So for sequence:

```
aggagtaagcccttgcaactggaaatacacccattg
```

The output would be something like:

```
a 12
g 8
t 7
c 9
```

for single bases

```
ag 1
ga 1
gt 1
aa 3
gc 2
cc 2
tt 1
ct 1
gg 1
at 2
ac 2
tg 1
```

for di-nucleotides

```
agg 1
agt 1
aag 1
ccc 1
ttg 2
caa 1
ctg 1
gaa 1
ata 1
cac 1
cca 1
```

for tri-nucleotides

Hint – the above examples only count what is present so there are no counts of zero. For bonus points find all possible nucleotide combinations in advance then count those that are present. You can also provide warnings about incomplete groups of nucleotides at the end of the sequence, e.g. if there are two bases at the end when you are counting tri-nucleotides, and deal with gaps or non-standard bases in the input sequence.

Challenge: find the single, di-nucleotide and tri-nucleotide counts for:

```
GAACCCGAAAATCCTTCCTTGCAGGAAACCAGTCTCAGTGTCCAACTCTCTAACCTTGGAACTGTGAGAA
CTCTGAGGACAAAGCAGCGGATACAACCTCAAAAGACGTCTGTCTACATTGAATTGGGATCTGATTCTTC
TGAAGATACCGTTAATAAGGCAACTTATTGCAGTGTGGGAGATCAAGAATTGTTACAAATCACCCCTCAA
GGAACCAGGGATGAAATCAGTTTGGATTCTGCAAAAAAGGCTGCTTGTGAATTTTCTGAGACGGATGTAA
```

2.6. Develop a function which gives the GC content of a sequence. This is the number of G plus C bases in a sequence as a percentage of the total number of bases in the sequence. So for the sequence:

```
aggagtaagcccttgcaactggaaatacacccattg
```

the GC content is 47.22%.

Hint: if you have completed 2.5 you can use that function to help achieve this.

Challenge: what is the GC content of:

```
GAACCCGAAAATCCTTCCTTGCAGGAAACCAGTCTCAGTGTCCAACTCTCTAACCTTGGAACTGTGAGAA
CTCTGAGGACAAAGCAGCGGATACAACCTCAAAAGACGTCTGTCTACATTGAATTGGGATCTGATTCTTC
TGAAGATACCGTTAATAAGGCAACTTATTGCAGTGTGGGAGATCAAGAATTGTTACAAATCACCCCTCAA
GGAACCAGGGATGAAATCAGTTTGGATTCTGCAAAAAAGGCTGCTTGTGAATTTTCTGAGACGGATGTAA
```