Python Unit 2 programming problem

September 22, 2016

These are the exercises for this week. YOU ONLY HAVE TO UPLOAD A TEXT FILE CONTAIN-ING THE SCRIPT TO SOLVE PROBLEM 2, problems 1 is optional and won't be marked.

Exercise 1. As you already know the method *string.count()*, will return the number of times the argument (word or letter) is contained in the string. For example: if DNA='ACGTACGTAGG' then DNA.count('A') returns 3 and DNA.count('T') will return 2. Write a program that mimic this behavior. That is, the user provides a DNA sequence and the nucleotide he wants to count and the program returns the total number of times that nucleotide appear in the DNA sequence. Tips:

1. As a first attempt write a program that only counts single nucleotides. Next, modify the program so that it counts any arbitrarily long of string of nucleotides occurring in your sequence.

2. Remember that strings behave as lists in many situations. For example, try typing the following (as it is written, in a single line) in an interactive python shell an hit enter (twice):

   >>> DNA='ACGTACGTAGG'

   >>>for n in DNA:print(n)

3. Note that the *range()* function returns a list and it (as any list) can be used as a collection of objects to interact with in for loops. For example, try typing the following (as it is written, in a single line) in an interactive python shell an hit enter (twice):

   >>>for k in range(10):print (k)

4. Remember that negative index return elements from a list starting from the end. Example:

   >>>DNA='ACGTACGTAGG'

   >>>DNA[-4]

   >>>'T'

5. Remember that the function *range()*, requires, at least one argument (upper limit of the interval), but can accept up to three: lower interval value, upper interval value and step. For example, try:

   >>>for k in range(1,10,2):print (k)

6. The *step* can be negative. Try:

   >>>for k in range(10,0,-1):print (k)

   ...

Exercise 2. Lets improve the script you submitted for the Unit 1 assignment using the tools learned in Unit2 (flow control): Write a program that ask the user for a DNA sequence (keyboard input) and do the following:

1. If the provided sequence is RNA, warn the user, but translate it to DNA and continue.

2. If the sequence provided by the user is not DNA or RNA (protein or other...) inform the user that it only accepts DNA sequences and stop the program.

3. Report detailed sequence statistics: sequence length, % of each base and % of each dinucleotide

4. Check if there are restriction sites for any of the following restriction enzymes: EcoRI (GAATTC), BamHI (GGATCC), HindIII (AAGCTT) and NotI (GCGGCCGC).

5. If any of these enzymes cut the DNA report ALL the sites where it cuts.