Rosalind problems journey

"Hey everyone! I'm excited to announce that I'm embarking on a coding journey and I want you to join me! Whether you're a beginner or an experienced coder, there's always something new to learn, and I'd love to have you along for the ride.

I'll be sharing coding problems, tips, and resources that I come across on my journey, and I'd love to hear about your experiences and discoveries as well. We can learn from each other and grow together as coders. So if you're interested in joining me on this journey, feel free to comment below or send me a message. Let's code together and make some amazing things!"

Of course, feel free to personalize this post to your own style and goals. Good luck with your coding journey and have fun!

Hey, if you are looking to start a career in bioinformatics, Rosalind problems is the best way to begin with. My favorite coder editor is PyCharm. If you are a beginner, use this link [Install PyCharm & Python on Windows 10 | Updated 2022 - YouTube](https://www.youtube.com/watch?v=XsL8JDkH-ec) to see how to install and start PyCharm code editor. I will help you to solve all the Rosalind problems and I will explain line by line.

**Problem 1**

A string is simply an ordered collection of symbols selected from some alphabet and formed into a word; the length of a string is the number of symbols that it contains.

An example of a length 21 DNA string (whose alphabet contains the symbols 'A', 'C', 'G', and 'T') is "ATGCTTCAGAAAGGTCTTACG."

Given: A DNA string s of length at most 1000 nt.

Return: Four integers (separated by spaces) counting the respective number of times that the symbols 'A', 'C', 'G', and 'T' occur in s

**Sample Dataset**

AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAAGAGTGTCTGATAGCAGC

**Sample Output**

20 12 17 21

Solution

***def count\_symbols(dna\_string):***

***symbol\_counts = {'A': 0, 'C': 0, 'G': 0, 'T': 0}***

***for symbol in dna\_string:***

***symbol\_counts[symbol] += 1***

***return symbol\_counts['A'], symbol\_counts['C'], symbol\_counts['G'], symbol\_counts['T']***

***dna\_string = “give your dataset “***

***counts = count\_symbols(dna\_string)***

***print(" ".join(str(count) for count in counts))***

Discussion

To solve this problem, we can loop through the DNA string character by character, and count the occurrence of each symbol. We can use a dictionary to store the counts for each symbol, with the keys being the symbols 'A', 'C', 'G', and 'T'. Here's some Python code that implements this approach:

The code here provided is a Python function that takes a DNA string as input and returns the count of each nucleotide symbol ('A', 'C', 'G', and 'T') in that string as a tuple of four integers.

Remmmeber -Dataset for DNA string will be given only when you start the solving the problem,ever time rosland wil cgange the DNA dataset.

* Start with defining a function called count\_symbols that takes a single parameter, dna\_string, which is the DNA string we want to count the nucleotide symbols in
* Create a dictionary called symbol\_counts that maps each nucleotide symbol to an initial count of zero. You can use this dictionary to keep track of how many times each symbol appears in the DNA string
* With the help of main loop function iterates over each character in the dna\_string, which represents a nucleotide symbol, and increments the corresponding count in symbol\_counts
* Returns a tuple of four integers: the counts of 'A', 'C', 'G', and 'T' symbols in the DNA string, respectively
* Here we defined a sample DNA string called dna\_string. We then call the count\_symbols function with dna\_string as an argument, and store the resulting tuple of counts in a variable called counts. Finally, we use a for loop and the join method to print the counts as a space-separated string.

