

# Introduction to the Bioinformatics Armory

Orr Shomroni

19 Oktober, 2022

## Problem

This initial problem is aimed at familiarizing you with Rosalind’s task-solving pipeline. To solve it, you merely have to take a given DNA sequence and find its nucleotide counts; this problem is equivalent to “Counting DNA Nucleotides” in the Stronghold.

Of the many tools for DNA sequence analysis, one of the most popular is the Sequence Manipulation Suite. Commonly known as SMS 2, it comprises a collection of programs for generating, formatting, and analyzing short strands of DNA and polypeptides.

One of the simplest SMS 2 programs, called **DNA stats**, counts the number of occurrences of each nucleotide in a given strand of DNA. An online interface for **DNA stats** can be found [here](#).

**Given:** A DNA string  $s$  of length at most 1000 bp.

**Return:** Four integers (separated by spaces) representing the respective number of times that the symbols ‘A’, ‘C’, ‘G’, and ‘T’ occur in  $s$ . Note: You must provide your answer in the format shown in the sample output below.

## Sample Dataset

```
AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC
```

## Sample Output

```
20 12 17 21
```

## Programming Shortcut

Our default choice for existing functions and modules to analyze biological data is BioPython, a set of freely available tools for computational biology that are written in Python. We will give you tips on how to solve certain problems (like this one) using BioPython functions and methods.

Detailed installation instructions for BioPython are available in PDF and HTML formats.

BioPython offers a specific data structure called **Seq** for representing sequences. **Seq** represents an extension of the “str” (string) object type that is built into Python by supporting additional biologically relevant methods like **translate()** and **reverse\_complement()**.

In this problem, you can easily use the built-in Python method **.count()** for strings. Here’s how you could count the occurrences of ‘A’ found in a **Seq** object.

```
from Bio.Seq import Seq
my_seq = Seq("AGTACACTGGT")
my_seq.count("A")
```

## 3

## Solution

```
from Bio.Seq import Seq

f=open("rosalind_ini.txt",'r')
text=f.readlines()[0].replace("\n","")
my_seq=Seq(text)

string=str(my_seq.count("A"))+" "+str(my_seq.count("C"))+" "+str(my_seq.count("G"))+" "+str(my_seq.count("T"))
print("Original string: "+my_seq
```

## Original string: CTTCTAAGCATTCTGTTTCGGCCCGGGCAGAAGAAGTTCTCGACTTAACTAGCCGGCCGAAATATTCCTGGGTGGTTGAT

```
print(string)
```

## 223 224 212 213