

BIFS 618: Java for Biotechnology

Dr. Alkharouf

Final Exam

Submission instructions:

Submit the source code (.java file(s)) for each question. Label them with the question number please.

Remember this is an individual exam, you may not seek help from anyone, or give help to anyone! That includes the use of online sites\blogs.

Q1. (30 pts) Design a Java application that has a GUI for sequence analysis. The GUI should allow the user to select a file containing a DNA sequence. And have the following functionality:

- 1- Calculate DNA composition (number of A, T, G and C's)
- 2- Calculate GC content (%)
- 3- Find Compliment
- 4- Find Reverse compliment

Design is completely up to you. But your application should have the functionality listed above. You could design buttons for each function, so when the user clicks on a function\button, the result will be shown in a text box below.

Q2. (20 pts) Write a program that reads in a whole genome (in FASTA format – see attached example) and computes the background codon frequencies. The background frequency of a codon is computed by the formula:

$$\text{background_frq}(\text{codon}) = 100 * N(\text{codon}) / \text{Total_codons}$$

where $N(\text{codon})$ is the number of occurrence of the codon across the entire genome, and Total_codons is the total number of all codons in the whole genome. Print out the background frequency of each codon, from AAA to TTT, to a file.

NOTE: To simplify the problem, just count codons that appear in reading frame 1. That should give approximately the same frequencies as in all six reading frames.

Q3. (20 pts) Look at the following partial class definition, and then respond to the questions that follow it:

```
public class PubMed
{
    private String title;
    private String author;
    private String journal;
    private int volume;
}
```

- a) Write a constructor for this class. The constructor should accept an argument for each field.
- b) Write accessor and mutator methods for each field.
- c) Draw a UML diagram for the class.
- d) Write a small PubMedDemo class, to construct a PubMed object and calls the methods you created.

Q4. (15 pts) Write a method named subArray that accepts an array of integers, a starting index, and a length as its parameters, and returns a new array containing the desired portion of the array. For example, if the following array is declared:

```
int[] array = {11, 5, 7, 12, -3, 4, 2, 19};
```

The method call subArray(array, 2, 4) should return the following new array:

```
{7, 12, -3, 4}
```

Q5. (15 pts) Write a method named countDuplicates that accepts a sorted array of integers as its parameter, and returns the number of duplicate elements in the array. For example, consider the following array:

```
int[] array = {1, 2, 3, 3, 8, 8, 8, 8, 11, 11, 11, 14, 19, 19};
```

The method call countDuplicates(array) would return 7, because there are 1 duplicate of 3, 3 duplicates of 8, 2 duplicates of 11, and 1 duplicate of 19.

Note that the array passed to your method is guaranteed to be in sorted order.