CS4054: Bioinformatics

Rushda Muneer

Assignment 2

Due Date: February 18,2025
(10 points)

Instructions: Complete the assignment independently, as it is an individual task. Any instances of plagiarism will be met with strict consequences. Ensure to submit coding assignments in **Python (Jupyter Notebook)** by the due date. Each challenge should be a separate cell in the notebook for readability. Carefully follow all the provided guidelines.

Code Challenge 1: Implement the Median String Algorithm which Identifies the median string of length k in a collection of longer strings. (5 points)

Input: An integer *k*, followed by a space-separated collection of strings *Dna*.
Output: A *k*-mer *Pattern* that minimizes *d*(*Pattern*, *Dna*) among all possible choices of *k*-mers. (If there are multiple such strings *Pattern*, then you may return any one.)

```
# Create your median_string function, along with any subroutines you
need
def median_string(dna: list[str], k: int) -> str:
    pass
```

Sample Input

```
3
AAATTGACGCAT GACGACCACGTT CGTCAGCGCCTG GCTGAGCACCGG AGTACGGGACAG
```

Sample Output

```
ACG
```

Code Challenge 2: Implement the Greedy Motif Search algorithm. (5 points)

Input: Integers *k* and *t*, followed by a space-separated collection of strings *Dna*.
Output: A collection of strings "*BestMotifs"* resulting from applying Greedy Motif Search.If at any step you find more than one *Profile*-most probable *k*-mer in a given string, use the one occurring first.

```
# Create your greedy_motif_search function, along with any subroutines
you need
def greedy_motif_search(dna: list[str], k: int, t: int) -> list[str]:
    pass
```

Sample Input

```
3 5
GGCGTTCAGGCA AAGAATCAGTCA CAAGGAGTTCGC CACGTCAATCAC CAATAATATTCG
```

Sample Output

```
CAG CAG CAA CAA CAA
```