BT-3051 Data Structures and Algorithms for Biology
Assignment - 5

Submission : Since this is a coding based assignment, students need to submit their codes in a zipped folder. Your zip file should be named something like BTyyBxxx.zip, based on your roll number. This zip file must contain the codes used for each of the problems in separate **.ipynb** files with proper documentation.

Deadline : 14 October 2024, 23:59 hrs
Bonus marks and penalty will be applied as mentioned in the course plan.

Instructions:
a. Total marks for this assignment - 30, 6 marks for each question.
b. No restrictions on usage of functions, unless otherwise specified.
c. Submissions will be evaluated only if they are provided in the required .ipynb format.

Question 1: **Football Field Navigation**

You are a football field manager, and you need to guide a robot across a training field represented by an (**mxn**) grid. The robot starts at the top-left corner of the field (i.e., grid[0][0]) and needs to reach the bottom-right corner (i.e., grid[m-1][n-1]) where the final training goal is located.

The field has damp spots(obstacles) and open spaces. Obstacles are marked as 1, and open spaces are marked as 0 in the grid. The robot can only move either **down** or **right** at any point in time, but it cannot move through obstacles.

Your task is to determine the total number of unique paths that the robot can take to reach the bottom-right corner, avoiding all obstacles.

Question 2: ***Optimal Overlap Alignment of DNA Strings***

Objective:
Given two DNA sequences, we want to find an optimal overlap alignment. In this type of alignment, we align a suffix of the first sequence with a prefix of the second sequence to maximize the alignment score. The overlap alignment should account for:

- ➔ Matching symbols (+1 score)
- ➔ Substitutions (-2 score)
- ➔ Linear gap penalties (2 points per gap)

Input Format:

You are given two DNA sequences, s, and t, in FASTA format. Each string has a length of at most 10,000 base pairs.

Task:

1. Find the score of the optimal overlap alignment and the alignment itself. Specifically:
2. Align a suffix of string s with a prefix of string t to maximize the alignment score.
3. The alignment score is calculated based on the following scoring scheme:
    - +1 for matching characters.
    - -2 for mismatches (substitutions).
    - -2 for gaps (insertions or deletions).
4. If multiple alignments yield the same optimal score, return any one of them.

Output Format:

1. First, output the optimal alignment score.
2. Then, output the aligned suffix of s and the aligned prefix of t to achieve this score.

Input :

>sequence1
CTAAGGGATTCCGGTAATTAGACAG
>sequence2
ATAGACCATATGTCAGTGACTGTGTAA

Output:

1
ATTAGAC-AG
AT-AGACCAT

Question 3: **Longest Increasing and Decreasing Subsequences in Gene Expression Levels**

In gene expression studies, researchers must activate certain genes during an experiment for specific time intervals. Each gene activation represents a critical experiment interval where the gene is expressed. However, due to resource limitations (e.g., using a specific tool to express genes), only one gene can be activated at a time.

Given a list of gene expression levels (or any ordered biological data, such as protein activity levels, nucleotide frequencies, etc.), we aim to find:

1. The longest subsequence where gene expression levels are increasing.
2. The longest subsequence where gene expression levels are decreasing.

Definitions:

1. A subsequence is a selection of gene expression levels from the data, taken in the same order they appear, but not necessarily consecutive.

2. A subsequence increases if each gene expression level exceeds the previous one.
3. A subsequence decreases if each gene expression level is less than the previous one.

Input:
A positive integer n (where $n \leq 10{,}000$), represents the number of observed gene expression levels.
A sequence of gene expression levels π of length n, which is a permutation of integers from 1 to n (representing relative levels of expression)

Task:
You need to return:

1. The longest increasing subsequence of gene expression levels.
2. The longest decreasing subsequence of gene expression levels.
3. If multiple subsequences of the same maximum length exist, return any one.

Output Format:
First, output the longest increasing subsequence of the gene expression levels.
Then, output the longest decreasing subsequence of the gene expression levels.

Sample Input :
5
5 1 4 2 3

Sample Output:
1 2 3
5 4 2

Question 4 : **The Wizard's Scroll and the Longest Palindrome**

In the mystical kingdom of Palindora, the royal wizards need your help to unlock a secret message hidden in an ancient scroll. The scroll contains a sequence of characters, and the message can only be revealed if the text forms a palindrome.(Assume the length of text is N and its given)

The wizards are willing to perform up to **k** magical substitutions, allowing them to change any character in the scroll to any other character to help form a palindrome.

Your task is to determine the length of the longest palindromic substring that can be obtained from the scroll after performing up to k substitutions.

Help the wizards reveal the longest palindromic message hidden in the scroll.

Question 5 : **Problem: DNA Sequence Matching Using KMP Algorithm**

In molecular biology, DNA sequences are often represented as strings consisting of four nucleotides: A (adenine), C (cytosine), G (guanine), and T (thymine). Scientists frequently need to search for a specific gene or motif within a longer DNA sequence. Given a reference DNA sequence and a target gene sequence, you are tasked with identifying whether the target gene appears in the reference sequence using the KMP algorithm for efficient string matching.

Input:

- A reference DNA sequence S of length n (e.g., a long strand of DNA).
- A target gene sequence T of length m (e.g., a shorter subsequence that may represent a gene).

Output:

Return the starting index (0-based) of all occurrences of the target gene T in the reference sequence S. If the target gene does not exist in the reference, return an empty list.

Thank You!