

CS 331: Algorithms and Complexity (Spring 2024)
Unique Number: 50930, 50935 50940, 50945

Assignment 5

Due on Tuesday, 5 March, by 11.59pm

Problem 1

(8 pts) Please keep solutions in this section to a single paragraph.

The SEQUENCE-ALIGNMENT problem gives us two strings and asks if there is an alignment that optimizes an objective function.

- (a) (6 pts) Give a naïve, brute-force algorithm to solve the problem and describe its runtime. Generate all possible alignments and check the corresponding cost.
- (b) (2 pts) Given $string_1 = \text{ALGO}$ and $string_2 = \text{TEST}$, and suppose the gap penalty $\delta = 1$ and the alignment costs are $C(x, y) = 1$ ($x \neq y$).
 1. What is the optimal alignment?
 2. What is the cost of the optimal alignment?

Problem 2

(12 pts) A string, w , is a palindrome if its reverse is exactly the same as w , e.g., “a” and “ana”. Writing a string as a *palindrome sum* is a decomposition of the string as contiguous sub-strings that are palindromes. For example, consider “coffee”. Here are ways to write it as a palindrome sum:

“c” + “o” + “f” + “f” + “e” + “e” (size = 6 substrings)
“c” + “o” + “f” + “f” + “ee” (size = 5 substrings)
“c” + “o” + “ff” + “e” + “e” (size = 5 substrings)
“c” + “o” + “ff” + “ee” (size = 4 substrings)

Notice that each substring is a palindrome and each sum concatenates to our word “coffee”.

The goal is to find the **minimum** number of sub-strings used to create a valid palindrome sum for any given string S . For example, the minimum number of sub-strings in our “coffee” example is 4.

For any input string, you can assume you have it stored in a one-dimensional array S (indexing S by 1). You can also assume that you have the function $isPali$, where $isPali(x, y)$ is true if the substring $S[x:y]$ (the inclusive substring from index x to y) is a palindrome.

- (a) Using dynamic programming, write the pseudo code of an iterative algorithm which solves the problem. Show the **time complexity** of your algorithm.
- (b) Prove the correctness of your algorithm.

Problem 3

(10 pts) Infinite Galactic Builders Inc. has a problem. They need to plan a company-wide party such that no employee and their direct manager will be invited. Each employee in the company is assigned a positive integer enjoyment grade. You’re given a tree hierarchy of the company where the CEO is the root and each employee is a child of their manager. You’re asked to prepare a list of employees that maximizes the sum of the enjoyment of the employees.

- (a) Show that greedily choosing employees according to enjoyment grade, will not work.
- (b) Describe an efficient dynamic programming algorithm that maximizes the sum of enjoyment grades of the employees. Show the **time complexity** of your algorithm. Prove the correctness of your algorithm.