

Problem Set 9

Due Date Nov / 10th / 2021
Name Michael Ghattas
Student ID 109200649
Collaborators Me, Myself & I

Contents

1	Instructions	1
2	Honor Code (Make Sure to Virtually Sign)	2
3	Standard 24- Dynamic Programming: Design DP Algorithms	3
3.1	Problem 2(a)	3
3.2	Problem 2(b)	4
3.3	Problem 2(c)	5

1 Instructions

- The solutions **must be typed**, using proper mathematical notation. We cannot accept hand-written solutions. Here's a short intro to \LaTeX .
- You should submit your work through the **class Canvas page** only. Please submit one PDF file, compiled using this \LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).
- You are welcome and encouraged to collaborate with your classmates, as well as consult outside resources. You must **cite your sources in this document**. **Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.
- You **must** virtually sign the Honor Code (see Section 2). Failure to do so will result in your assignment not being graded.

2 Honor Code (Make Sure to Virtually Sign)

Problem 1. • My submission is in my own words and reflects my understanding of the material.

- Any collaborations and external sources have been clearly cited in this document.
- I have not posted to external services including, but not limited to Chegg, Reddit, StackExchange, etc.
- I have neither copied nor provided others solutions they can copy.

I agree to the above, Michael Ghattas.

□

3 Standard 24- Dynamic Programming: Design DP Algorithms

Problem 2. The Highest Reflective Subsequence problem is defined as follows.

- Instance: A string $\omega \in \Sigma^n$, where Σ is our fixed alphabet.
- Solution: A subsequence $\psi \in \Sigma^\ell$ of ω of maximum length such that for all $i \in \{1, \dots, \ell\}$, $\psi_i = \psi_{\ell-i+1}$.

The goal of this problem set is to design a dynamic programming algorithm to solve the Highly Reflective Subsequence problem.

3.1 Problem 2(a)

- (a) Let $L[i, j]$ denote the length of the highest reflective subsequence of $\omega[i, \dots, j]$. Write down a mathematical recurrence for $L[i, j]$. Clearly justify each case.

Answer:

$$L[i, j] = \begin{cases} 0 & : i > j \\ 1 & : i = j \\ 2 + L[i + 1, j - 1] & : \omega_i = \omega_j, i < j \\ \max(L[i + 1, j], L[i, j - 1]) & : \omega_i \neq \omega_j, i < j \end{cases}$$

□

3.2 Problem 2(b)

- (b) Clearly describe how to construct and fill in the lookup table. For the cell $L[i, j]$, clearly describe the sub-cases we consider, which optimal sub-case we select, and any relevant pointers that should be included in the table.

Answer:

We will utilize a two-dimensional array to store the values of our cases:

- **Case 1:** All cell entry where the value of $i > j$ will be filled with 0
- **Case 2:** All diagonal cell entry where the value of $i = j$ will be filled with 1
- **Case 3:** All cell entries where we find the value of $\omega_i = \omega_j$ and $i < j$ will be filled with $2 + L[i + 1, j - 1]$, while pointing to the cell entry containing $L[i + 1, j - 1]$.
- **Case 4:** All cell entries where the value of $\omega_i \neq \omega_j$ and $i < j$ will be filled with $\max(L[i + 1, j], L[i, j - 1])$, while pointing to the cell entry containing the *max* of $L[i + 1, j]$ or $L[i, j - 1]$.

□

3.3 Problem 2(c)

- (c) Work through an example of your algorithm using the input string $\omega = \text{abcdeca}$. Clearly show how to recover an optimal solution using your lookup table. You may hand-draw your lookup table, but your explanation must be typed.

Answer:

First we note that $\text{length}(\omega) = 7$, and proceed:

- $L[7, 1] = L[7, 2] = L[7, 3] = L[7, 4] = L[7, 5] = L[7, 6] = L[6, 1] = L[6, 2] = L[6, 3] = L[6, 4] = L[6, 5] = L[5, 1] = L[5, 2] = L[5, 3] = L[5, 4] = L[4, 1] = L[4, 2] = L[4, 3] = L[3, 1] = L[3, 2] = L[2, 1] = 0$
- $L[1, 1] = L[2, 2] = L[3, 3] = L[4, 4] = L[5, 5] = L[6, 6] = L[7, 7] = 1$
- $L[1, 2] = \max(L[2, 2], L[1, 1]) = 1$
- $L[2, 3] = \max(L[3, 3], L[2, 2]) = 1$
- $L[1, 3] = \max(L[2, 3], L[1, 2]) = 1$
- $L[3, 4] = \max(L[4, 4], L[3, 3]) = 1$
- $L[2, 4] = \max(L[3, 4], L[2, 3]) = 1$
- $L[1, 4] = \max(L[2, 4], L[1, 3]) = 1$
- $L[4, 5] = \max(L[5, 5], L[4, 4]) = 1$
- $L[3, 5] = \max(L[4, 5], L[3, 4]) = 1$
- $L[2, 5] = \max(L[3, 5], L[2, 4]) = 1$
- $L[1, 5] = \max(L[2, 5], L[1, 4]) = 1$
- $L[5, 6] = \max(L[6, 6], L[5, 5]) = 1$
- $L[4, 6] = \max(L[5, 6], L[4, 5]) = 1$
- $L[3, 6] = 2 + L[4, 5] = 2 + 1 = 3$
- $L[2, 6] = \max(L[3, 6], L[2, 5]) = L[3, 6] = 3$
- $L[1, 6] = \max(L[2, 6], L[1, 5]) = L[2, 6] = 3$
- $L[6, 7] = \max(L[7, 7], L[6, 6]) = 1$
- $L[5, 7] = \max(L[6, 7], L[5, 6]) = 1$
- $L[4, 7] = \max(L[5, 7], L[4, 6]) = 1$
- $L[3, 7] = \max(L[4, 7], L[3, 6]) = L[3, 6] = 3$
- $L[2, 7] = \max(L[3, 7], L[2, 6]) = L[2, 6] = 3$
- $L[1, 7] = 2 + L[2, 6] = 2 + 3 = 5$

$L[i, j]$	1	2	3	4	5	6	7
1	1	$\leftarrow 1$	$\leftarrow 1$	$\leftarrow 1$	$\leftarrow 1$	$\downarrow 3$	$\swarrow 5$
2	0	1	$\leftarrow 1$	$\leftarrow 1$	$\leftarrow 1$	$\downarrow 3$	$\leftarrow 3$
3	0	0	1	$\leftarrow 1$	$\leftarrow 1$	$\swarrow 3$	$\leftarrow 3$
4	0	0	0	1	$\leftarrow 1$	$\leftarrow 1$	$\leftarrow 1$
5	0	0	0	0	1	$\leftarrow 1$	$\leftarrow 1$
6	0	0	0	0	0	1	$\leftarrow 1$
7	0	0	0	0	0	0	1

Let ψ be the Highly Reflective Subsequence of ω . From the table we can see that the optimal string length (5) for our solution is located in cell L[1, 7] representing $\psi_1 = \psi_5 = \text{a}$. Starting from there and following our pointers we move to cell L[2, 6] which leads us to cell L[3, 6] representing $\psi_2 = \psi_4 = \text{c}$. Finally from there we follow the pointers to cell L[4, 5] which leads us to cell L[4, 4] representing $\psi_3 = \text{d}$. Thus our final solution is $\psi = \text{acdca}$. \square