Quiz 25 Standard 25 – DP – Design a DP Algorithm

Due Date ....TODO
Name ....**Blake Raphael**
Student ID **109752312**
Quiz Code (enter in Canvas to get access to the LaTeX template) .... **GHSDA**

Contents

Instructions

- You may either type your work using this template, or you may handwrite your work and embed it as an image in this template. **If you choose to handwrite your work, the image must be legible, and oriented so that we do not have to rotate our screens to grade your work.** We have included some helpful LaTeX commands for including and rotating images commented out near the end of the LaTeX template.

- You should submit your work through the **class Gradescope 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 **may not collaborate with other students**. **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, Discord, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.

- You **must** virtually sign the Honor Code (see Section). Failure to do so will result in your assignment not being graded.

Honor Code (Make Sure to Virtually Sign)

Problem HC.
- 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.

Agreed (I agree to the above, Blake Raphael). □

25 Standard 25: Dynamic Programming: Design a Dynamic Programming Algorithm

Problem 25. Here we consider a weighted version of the string alignment (aka edit distance) problem from class. In this version, each operation has a different cost:

- The cost of a delete operation is 2

- The cost of an insert operation is 3

- The cost of an substitute operation 5

- (No-op is still free, when the letters match)

The Weighted Edit Distance problem is:

- Input: Two strings x, y

- Output: The minimum total cost of a sequence of delete, insert, and substitute operations that transforms x into y.

Complete the following parts.

(a) We are given the following **subproblem**: Define $t[n, m]$ to be the minimum cost of delete, insert, and substitute operations needed to transform the prefix $x[1..n]$ into the prefix $y[1..m]$ (note that when $n = 0$, the prefix $x[1..n]$ is, by definition, the empty string).

Identify the **recurrence relation** for the problem: (2 points)

Answer.
$$t[n, m] = \begin{cases} 0 & : n = 0, m = 0 \\ t[n-1, m-1] & : \text{if } x_n = y_m \\ 2 + min(t[n-1, m], t[n, m-1]) & : \text{if } n < m \text{ and } x_n \neq y_m \\ 3 + min(t[n-1, m], t[n, m-1]) & : \text{if } n > m \text{ and } x_n \neq y_m \\ 5 + min(t[n-1, n], t[n, m-1]) & : \text{if } x_n \neq y_m \text{ and } n = m \end{cases}$$

□

(a) Let $x = cat$ and $y = kite$. Using a bottom-up dynamic programming approach, determine the minimum-cost sequence of delete, insert, and substitution operations to transform x into y. For your convenience, we have provided the lookup table for you to use. You may also hand-draw your lookup tables.

Fill in the entire table; clearly show all work for how you filled in **row 3 (corresponding to $x[1..3] = cat$)** of the table (you don't have to show work for the rest of the table, but you **do** need to fill it in). (2 points)

Answer. The final DP table is as follows:

	-	k	i	t	e
-	0	3	6	9	12
c	3	5	8	11	14
a	6	8	10	13	16
t	9	11	13	10	12

Work for row 3 (corresponding to the prefix $x[1..3] = cat$)):

- (3,0): 9 because you have to have 3 inserts. $3 \cdot 3 = 9$.

- (3,1): 11 because you need 1 substitution and 2 inserts. $5 + (3 \cdot 2) = 11$.

- (3,2): 13 because you need 2 substitutions and 1 insert. $(5 \cdot 2) + 3 = 13$.

- (3,3): 10 because you need 2 substitutions and 1 no-op. $(5 \cdot 2) + 0 = 10$.

- (3,4): 12 because you need 2 substitutions, 1 no-op, and 1 deletion. $(5 \cdot 2) + 0 + 2 = 12$.

$\square$