

CSC210 Advanced Algorithm and Design Lab
20/02/2023

Time: 1 Hour
Marks: 150

Instructions

1. Write the programs with proper comments and indentation
2. Create a directory <Admission Number>_<Date> [21JEXXXX_090122], copy all the files into it and **upload in Google Class Room**
3. Submit a single C/C++ source file
4. Do not use STL calls
5. Each program should start with these comment lines:

/*

Name:

ID No:

*/

Q1. We want to measure the minimum edit distance between two strings. An edit is defined by insertion, deletion, or replacement of a character. The distance equation is given as follows:

$$lev_{x,y}^{(i,j)} = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0 \\ \min \begin{cases} lev_{x,y}^{(i-1,j)} + 1 \\ lev_{x,y}^{(i,j-1)} + 1 \\ lev_{x,y}^{(i-1,j-1)} + 1_{(a_i \neq b_j)} \end{cases} \end{cases}$$

Write a dynamic programming algorithm to find edit distance between two strings. The two strings might be of different length. x and y represent two strings and i, j represents the character index.

Write a function *Compute_Distance* that takes two strings as input and print the distance and the changes applied across each index i.e., the sequence of changes that lead to the cost value.

The **main()** function:

1. Take two strings from user and call the function.[5]

2. Call *Compute_Distance* to find out the edit distance and sequence of changes between two strings [35+35]

Example:

1. kitten -> sitten (substitution of "s" for "k")
2. sitten -> sittin (substitution of "i" for "e")
3. sittin -> sitting (insertion of "g" at the end)

Edit distance: 3

Q2. A student is registering for courses and can take a maximal credit of N. There are n courses available in the semester and credit of ith course is ci and its utility is ui. What courses should the student take?

In this context, the courses should be selected in such a way that the student will register for those courses for which he will gain maximum utility. Hence, the objective of the student is to maximize the utility.

Write a function *Binary_DP_Register* that take the credit, utility, and course details as input and find out courses based on dynamic programming. In *Binary_DP_Register* a student can either take or drop a course. Print the courses, and total utility obtained in this approach.

The **main()** function:

1. Take input (number of courses n) from user. Take the corresponding credit (c) and utility (u) for each of the courses and maximal credit N. [10]
2. Call *Binary_DP_Register* to find out the optimal solution [65]

Example

Course	A	B	C	D
Utility	280	100	120	120
Credit	40	10	20	24

N = 60

Actual Optimal: A, C = 280 + 120 = 400