CS21003
Tutorial 3
August 11th, 2017

1. Given a sequence of n real numbers *A(1)…A(n)*, determine a contiguous subsequence *A(i) ...A(j)* for which the sum of elements in the subsequence is maximized. E.g., for input {3,-4,9,-8,8,7}, the output will be {9, -8, 8, 7} with sum = 16. Propose a dynamic programming algorithm for this problem and show the working of your algorithm using the example provided.

2. Suppose you are given 2 strings, str1 and str2 of lengths ***n*** and ***m***, respectively. Your task is to find the length of the longest subsequence common to both str1 and str2 using DP. E.g., for input str1="classical", str2="musical", output with be subsequence "sical" with length=5. What would the complexity of the algorithm? Show the working of your algorithm on this example.

3. Using the solution of problem 2, try to find the minimum number of characters to be inserted to a given string str to convert it to a palindrome. E.g., for an input str ABCDE, output should be 4 (characters to be inserted with the corresponding string: ABCDEDCBA).

4. 7. A sequence is called a good sequence if $a_1 < a_2 > a_3 < a_4 \ldots a_k$. i.e. $a_i < a_{i+1}$ if i is odd and $a_i > a_{i+1}$ if i is even for all i < k. You are given a sequence A containing n integers. You need to find the length of longest good subsequence of A using dynamic programming algorithm. For example, for the input sequence {1,2,6,5,3,4} the largest such sequence is {2,6,3,4}. Show the working of your algorithm using this example.

5. You are given a set A of n positive integers and a value ***sum***. Your task is to determine whether or not there exists any subset of the given set A, the sum of whose elements is equal to the given value of ***sum***. For example, given A = {2,4,6,9} and ***sum*** =17, you can find the subset {2,6,9} that adds up to sum and the algorithm should return *True*. Propose a DP algorithm for this problem and show its working using the above example.

6. You are given n identical dices, where each dice has a given number of faces m, and these faces are labeled with {1,2,...m}. Suppose you throw all the dices together and compute the sum as per the faces that show up on the throw. Given a sum, your task is to find in how many ways can you get the given sum? For example, for n=2 (number of dices), m=3 (each dice having three faces {1,2,3}) and sum=4 (required sum of faces), there are 3 ways in which you can obtain this sum (1+3, 2+2, 3+1). Propose a dynamic programming solution for this problem and show its working using this example.

7. Given weights and values of n items, you have to put these items in a knapsack of capacity M to get the maximum total value in the knapsack. Show how your algorithm works using the following example. Number of items: n=4, weight of items, w={2 3 4 5}, value of items: v={3 4 5 6}, capacity of knapsack: M=5.