

Homework 10 : Algorithms and Data Structures

Digdarshan Kunwar

May 2019

!! NOTE !!

Implementation of different algorithms are in different folders.

Problem 10.1

Longest Ordered Sub-array

- (a) The solution is within the folder: Longest Ordered Subarray

The implementation is in LOS.py

\$: make LOS

or

\$: make all

Problem 10.2

Sum in triangles

- (a) The solution is within the folder:Sum In Triangles

The implementation is in sumofTriangles.py

\$: make triangle

or

\$: make all

- (b) The brute force approach means that the algorithm would try out all the possible path so the run time would be like 2^n but in the dynamic approach that we implemented from the bottom up approach we have a running time of n^2 . So in general it is better to use dynamic programming approach.
- (c) Here the a greedy algorithm does not work for this problem. This is because in the greedy approach the algorithm gets the max element in that particular level starting from the top which may not maintain the path of the triangle to which we have a maximum sum. Thus it may break the path in which the max-sum is being generated, therefore Greedy algorithm is a proper solution to this problem. The problem is that the path generated by Greedy algorithm is not necessarily the optimal solution for the problem.

Problem 10.3

Scuba Diver

- (a) The solution is within the folder: Scuba Diver

```
The implementation is in scuba.cpp
$ : make all
$ : make run
$ : make clean
```

References :

Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (n.d.). Introduction to algorithms.
<https://www.geeksforgeeks.org/longest-increasing-subarray/>
<https://www.geeksforgeeks.org/longest-increasing-subsequence-dp-3/>
<https://www.geeksforgeeks.org/maximum-path-sum-triangle/>
<https://www.spoj.com/problems/SCUBADIV/> <http://codeforces.com/blog/entry/6480>