

Algorithms and Data Structures

Drishti Maharjan

May 3, 2019

Assignment 10

Problem 10.1

Find code in 10.1.cpp

Problem 10.2

a. Code in 10.2a.cpp

b. For the dynamic programming implementation I have used for problem 10.2 a, let's ignore the input part, and analyze only the part where dynamic program concept is implemented after finding cn (no of lines in the triangle). Consider from line 132. I'll refer to cn as n in this analysis. I have 3 separate loops in total, with each having a nested loop. Considering the worst case, each loop lump section will loop $(n * n)$ times, given its inner loop. So, adding up all the lumps, we get $3(n * n)$. This is equivalent to $O(n^2)$. However, also note that in our general cases, all the loops will loop lesser than $(n * n)$, since our nested loop structure is $i = 0$ to n , and $j = 0$ to i .

In brute force approach, we check every path and see which gives the maximum value. For every point in the triangle except in the last line, there are 2 possible paths for it to take. So, the time complexity would be $O(2^n)$. From this, we can clearly say that dynamic programming approach is way faster than brute force approach.

c. One very significant drawback of greedy algorithm is that while it ensures optimal solution at each level, it can't ensure a global optimal for the overall solution. For this question, greedy approach would, in each point compare 2 values at the left or right and then select that path. But this can't always guarantee we get the globally optimal path. For example, in the test case mentioned in the question, the followed path would be 7-8-1-7-5, and the result would be 28, even though the most optimal solution should be 30.

Problem 10.3

Code in 10.3.cpp