Problem 10.1

The code implementation is in 10-1.cpp.

Problem 10.2

a) The code implementation is in 10-2.cpp.

b) Brute-force approach for finding the path for maximum sum

The time complexity for this problem with dynamic programming is just $O(n^2)$. We add the larger element from the row below to each element in the current row by comparing the direct below and right below elements of each element in the bottom-up approach. Finally, the very first element in the first row will be with the maximum sum. Hence, we don't need to look back to the elements in the row below after each iteration.

The outer loop will iterate for n-1 times, and the inner iterates from 0 to i (iterator of outer loop) times. So, the total number of operations would be $[(n-1)n]/2 = [n^2-n]/2$.

Let $f(n) = [n^2-n]/2$, $g(n) = n^2$, $c = 0.25$, $cg(n) = 0.25n^2$

So, $0 \leq f(n) \leq cg(n)$ if $n \geq 2$

Hence, time complexity is $O(n^2)$.

If we use Brute Force approach, the time complexity would be around $O(2^{(n)})$, using the number of bits of each number to compare. It will take a lot more time than the dynamic programming approach even for small n lines in the triangle.

c) A greedy algorithm cannot always give the globally optimal solution. Greedy algorithm will not check and compare all possible paths to produce the real maximum sum. For example, the greedy algorithm would just choose the larger element among two adjacent below elements to be added to the current element in the above row, and add to the first element finally. It is not guaranteed whether it would be the most accurate choice. So, greedy algorithm should not be used.

Problem 10.3

The code implementation is in 10-3.cpp.

References

Goel, A (2017, March 21). Construction of Longest Increasing. Retrieved from https://www.geeksforgeeks.org/construction-of-longest-increasing-subsequence-using-dynamic-programming/
(I used this website to get some inspirations for problem 10.1.)

Jacobs.W, Mason.A, Pann.S. Maximum Path Sum: An optimization problem. Retrieved from http://people.uncw.edu/tagliarinig/Courses/380/F2017%20papers%20and%20presentations/Shanghai%20Knights--Jacobs,%20Mason,%20and%20Pann/Presentation.pptx
(I used this website to get some inspirations for problem 10.2c.)

Laituan245 (2015, 23 August). SCUBADIV - Scuba diver. Github. Retrieved from https://github.com/laituan245/spoj-solution/blob/master/classical/SCUBADIV.cpp.
(I used this website to get some inspirations for problem 10.3.)

Mathblog.dk (2017, December 23). Project Euler 18: Maximum Sum in the Triangle in C# | MathBlog.
Retrieved from https://www.mathblog.dk/project-euler-18/
(I used this website to get some inspirations for problem 10.2b.)

Pradhan, S(2018, May 23).Maximum path sum in a triangle. Retrieved from https://www.geeksforgeeks.org/maximum-path-sum-triangle/
(I used this website to get some inspirations for problem 10.2a.)