1. Recursion

1. Subset – I | Sum Of all Subsets (pick or not pick)

<https://takeuforward.org/data-structure/subset-sum-sum-of-all-subsets/>

2. Subset – II | Print all the Unique Subsets (pick from any index and add check for not picking duplicate elements)

<https://takeuforward.org/data-structure/subset-ii-print-all-the-unique-subsets/>

3. Combination Sum – 1 (pick or not pick) stay on same index

<https://takeuforward.org/data-structure/combination-sum-1/>

4. Combination Sum II – Find all unique combinations (pick from any index and add check for not picking duplicate sets)

<https://takeuforward.org/data-structure/combination-sum-ii-find-all-unique-combinations/>

5. Palindrome partitioning (pick from any index)

<https://takeuforward.org/data-structure/palindrome-partitioning/#google_vignette>

6. Climbing Stairs:

<https://takeuforward.org/data-structure/dynamic-programming-climbing-stairs/>

2. Recursion And Backtracking

1. Print All Permutations of a String/Array

<https://takeuforward.org/data-structure/print-all-permutations-of-a-string-array/>

(check solution at

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\PermutationOfArray\RecusriveApproach.java)

**3. Dynamic Programming**

1. Fibonacci series using DP:

<https://takeuforward.org/data-structure/dynamic-programming-introduction/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\fibonacciSeries\_DP

2. Climbing Stair :

<https://takeuforward.org/data-structure/dynamic-programming-climbing-stairs/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\ClimbingStairs

3. Frog Jump :

<https://takeuforward.org/data-structure/dynamic-programming-frog-jump-dp-3/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\frog\_Jump

4. Maximum Sub of non-adjacent element:

<https://takeuforward.org/data-structure/maximum-sum-of-non-adjacent-elements-dp-5/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MaxSumOfNonAdjacentElements

5. House Robber

<https://takeuforward.org/data-structure/dynamic-programming-house-robber-dp-6/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\HouseRobber\_2\_DP

6.Ninja Training:

<https://takeuforward.org/data-structure/dynamic-programming-ninjas-training-dp-7/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Ninja\_Training\_DP

7. Grid Unique Paths :

<https://takeuforward.org/data-structure/grid-unique-paths-dp-on-grids-dp8/>

8.Grid Unique Paths – variation 2

<https://takeuforward.org/data-structure/grid-unique-paths-2-dp-9/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\GridUniquePaths\_2\_DPonGrids

9. Minimum path sum in Grid :

<https://takeuforward.org/data-structure/minimum-path-sum-in-a-grid-dp-10/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MinimumPathSum\_In\_Grid\_DPonGrids

10. Minimum Path Sum in Triangular Grid :

<https://takeuforward.org/data-structure/minimum-path-sum-in-triangular-grid-dp-11/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MinimumPathSum\_In\_TriangulerGrid\_DPonGrids

11. Minimum / maximum Falling Path Sum :

<https://takeuforward.org/data-structure/minimum-maximum-falling-path-sum-dp-12/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MinimumFallingPathSum\_DPonGrids

12. Subset Sum equal to Target:

<https://takeuforward.org/data-structure/subset-sum-equal-to-target-dp-14/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SubSetSum\_EqualTo\_Target

13. Partition Equal Subset Sum :

<https://takeuforward.org/data-structure/partition-equal-subset-sum-dp-15/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\PartitionSum\_EqualTo\_Target

14. Count Subset with Sum K :

<https://takeuforward.org/data-structure/count-subsets-with-sum-k-dp-17/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\CountSubsetsWith\_SumK\_DPOnSubsets

15. Count Partitions with Given Difference :

<https://takeuforward.org/data-structure/count-partitions-with-given-difference-dp-18/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\CountPartitions\_With\_GivenDifference\_DP

1. Longest Increasing Subsequence (take or not take and return max length)

<https://takeuforward.org/data-structure/longest-increasing-subsequence-dp-41/>

Check Solution of recursive and Memoization DP at below location:

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LongestIncreasingSubseq\_DP

2. Longest Common Subsequence

<https://takeuforward.org/data-structure/longest-common-subsequence-dp-25/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LongestCommon\_Subseq\_DP

3. 0/1 Knapsack

<https://takeuforward.org/data-structure/0-1-knapsack-dp-19/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Zero\_One\_knapsack

4. Coin Change

<https://takeuforward.org/data-structure/coin-change-2-dp-22/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\CoinChange2\_DP

5. Rod Cutting Problem

<https://takeuforward.org/data-structure/rod-cutting-problem-dp-24/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\sr c\Rod\_Cutting\_Problem\RecursiveApproach.java

**3. Array Problems**

1. Set Matrix Zeros:

<https://takeuforward.org/data-structure/set-matrix-zero/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SetMatrixZero.java

2. Kadane’s Algorithm : Maximum Subarray Sum in an Array

<https://takeuforward.org/data-structure/kadanes-algorithm-maximum-subarray-sum-in-an-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MaximumSubarraySum.java

3. Sort an array of 0s, 1s and 2s

<https://takeuforward.org/data-structure/sort-an-array-of-0s-1s-and-2s/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SortArrayOfZeroOneTwo\_InPlace.java

4. Stock Buy And Sell

<https://takeuforward.org/data-structure/stock-buy-and-sell/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Stock\_Buy\_And\_Sell.java

5. Merge two Sorted Arrays Without Extra Space

<https://takeuforward.org/data-structure/merge-two-sorted-arrays-without-extra-space/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MergeTwoSortedArray\_withoutExtraSpace.java

6. Find the duplicate in an array of N+1 integers

<https://takeuforward.org/data-structure/find-the-duplicate-in-an-array-of-n1-integers/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\FindDuplicateInArray.java

7. Find the repeating and missing numbers

<https://takeuforward.org/data-structure/find-the-repeating-and-missing-numbers/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\FindMissingNumberInArray.java

8. Search in a sorted 2D matrix

<https://takeuforward.org/data-structure/search-in-a-sorted-2d-matrix/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SearchElementIn2DMatrix.java

9>Grid Unique Paths | Count paths from left-top to the right bottom of a matrix

<https://takeuforward.org/data-structure/grid-unique-paths-count-paths-from-left-top-to-the-right-bottom-of-a-matrix/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\GridUniquePaths\_DPonGrid

10>Two Sum : Check if a pair with given sum exists in Array

<https://takeuforward.org/data-structure/two-sum-check-if-a-pair-with-given-sum-exists-in-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\TwoSum.java

11>Longest Consecutive Sequence in an Array

<https://takeuforward.org/data-structure/longest-consecutive-sequence-in-an-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LongestConsecutiveSequenceInArray.java

12>Spiral Traversal of Matrix

<https://takeuforward.org/data-structure/spiral-traversal-of-matrix/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SpiralMatrix.java

13>Leaders in an Array

<https://takeuforward.org/data-structure/leaders-in-an-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LeadersInArray.java

14>Union of Two Sorted Arrays

<https://takeuforward.org/data-structure/union-of-two-sorted-arrays/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\UnionOfTwoSortedArray.java

15>Find the number that appears once, and the other numbers twice

<https://takeuforward.org/arrays/find-the-number-that-appears-once-and-the-other-numbers-twice/>

16>Find the missing number in an array

<https://takeuforward.org/arrays/find-the-missing-number-in-an-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\FindMissingNumberInArray.java

17>Remove Duplicates in-place from Sorted Array

<https://takeuforward.org/data-structure/remove-duplicates-in-place-from-sorted-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\Arrays\_Problems

18>Rotate array by K elements

<https://takeuforward.org/data-structure/rotate-array-by-k-elements/>

19>Move all Zeros to the end of the array

<https://takeuforward.org/data-structure/move-all-zeros-to-the-end-of-the-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\MoveZerosToEndOfAnArray.java

20>Count Maximum Consecutive One’s in the array

<https://takeuforward.org/data-structure/count-maximum-consecutive-ones-in-the-array/>

21>Check if the given String is Palindrome or not

<https://takeuforward.org/data-structure/check-if-the-given-string-is-palindrome-or-not/>

22>Check for Balanced Parentheses

<https://takeuforward.org/data-structure/check-for-balanced-parentheses/>

23>Max Product Sub array

<https://takeuforward.org/data-structure/maximum-product-subarray-in-an-array/>

24>Check Anagrams

<https://takeuforward.org/data-structure/check-if-two-strings-are-anagrams-of-each-other/>

25> refer String problems from   
<https://takeuforward.org/interviews/tcs-nqt-coding-sheet-tcs-coding-questions/#point_6>

**4 .Two Pointers & Sliding Window Problem**

1. Longest Substring without repeating Character :

<https://takeuforward.org/data-structure/length-of-longest-substring-without-any-repeating-character/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\LongestSubstringWithoutRepeatingChars.java

1. Max Consecutive One’s part 3:

<https://www.youtube.com/watch?si=SoOW64pP6otEKxBw&v=3E4JBHSLpYk&feature=youtu.be>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\MaxConsecutiveOnesPart3.java

1. Longest Repeating Character in String Replacement :

<https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\LongestRepeatingCharsReplacement.java

1. Binary Subarray with Sum

<https://www.youtube.com/watch?v=XnMdNUkX6VM>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\BinarySubArrayWithSum.java

1. Count No Of Nice Sub Arrays:

<https://www.youtube.com/watch?v=j_QOv9OT9Og>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\CountNoOfNiceSubArrays.java

1. Number Of Substring Containing All 3 characters :

<https://www.youtube.com/watch?v=xtqN4qlgr8s>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\NumberOfSubstringContainingAll3Chars.java

1. Max Points you can obtain from Cards :

<https://www.youtube.com/watch?v=pBWCOCS636U>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\MaxPointsFromCard.java

1. Longest Substring at the most K characters :

<https://www.youtube.com/watch?si=Kh0_u6aCkkBU3Q33&v=teM9ZsVRQyc&feature=youtu.be>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\SlidingWindow\_TwoPointers\_Problems\LongestSubstringWithKDistinctChars.java

**Stack & Queue Implementations:**

1. Implement Stack using Array :

<https://takeuforward.org/data-structure/implement-stack-using-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\StackUsingArrayImpl.java

1. Implement Queue Using Array :

<https://takeuforward.org/data-structure/implement-queue-using-array/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\QueueUsingArrayImpl.java

1. Implement Stack using Linked List:

<https://takeuforward.org/data-structure/implement-stack-using-linked-list/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\StackUsingLinkedList.java

1. Implement Queue using Linked List :

<https://takeuforward.org/data-structure/implement-queue-using-linked-list/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\QueueUsingLinkedList.java

1. Implement Stack using Queue :

<https://takeuforward.org/data-structure/implement-stack-using-single-queue/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\StackUsingQueue.java

1. Implement Queue using Stack :

<https://takeuforward.org/data-structure/implement-queue-using-stack/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\QueueUsingStackImple.java

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\StackAndQueueImplementation\QueueUsingStackImpl\_Approach2.java

**Linked List Problems:**

1. Delete Start Node from Linked List :

<https://www.youtube.com/watch?v=VaECK03Dz-g&t=1934s>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LinkedList\DeleteFirstNodeFromLinkedList.java

1. Delete End Node from Linked List :

<https://takeuforward.org/data-structure/delete-last-node-of-linked-list/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LinkedList\DeleteLastNodeFromLinkedList.java

1. Delete kth node from Linked List:

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LinkedList\DeleteKthNodeFromLinkedList.java

1. Insert a node into Linked list

(at the head , at the tail , at the given position , before a given node value)

<https://takeuforward.org/linked-list/insert-at-the-head-of-a-linked-list>

<https://www.youtube.com/watch?si=vHSwdf9jhE05adKM&t=1934&v=VaECK03Dz-g&feature=youtu.be>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\LinkedList\InsertNodeIntoLinkedList.java

1. Find the Length Of Linked List : Refer above program
2. Search an Element in Linked List: Refer above program

**Greedy Algorithm problems:**

1. Assign Cookies:

<https://takeuforward.org/Greedy/assign-cookies>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\AssignCookies.java

1. Fractional Knapsack :

<https://takeuforward.org/data-structure/fractional-knapsack-problem-greedy-approach/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\FractionalKnapsack.java

1. Find Minimum No of coins:

<https://takeuforward.org/data-structure/find-minimum-number-of-coins/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\FindMinimumNoOfCoins.java

1. Lemonade change :

<https://takeuforward.org/Greedy/lemonade-change>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\LemonadeChange.java

1. Valid Paranthesis checker :

<https://www.youtube.com/watch?si=XRHeyh7jOaLaTy3g&v=cHT6sG_hUZI&feature=youtu.be>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\ValidParanthesis\_Recursive.java

1. N Meetings in One Room :

<https://takeuforward.org/data-structure/n-meetings-in-one-room/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\NMeetingsinOneRoom.java

1. Jump Game :

<https://takeuforward.org/Greedy/jump-game-i>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\JumpGame.java

1. Minimum No of platforms required for Railway:

<https://takeuforward.org/data-structure/minimum-number-of-platforms-required-for-a-railway/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\MinimumNoOfPlatforms.java

1. Job Sequencing Problem :

<https://takeuforward.org/data-structure/job-sequencing-problem/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\JobSequencingProblem.java

1. Program to find Shortest Job First :

<https://takeuforward.org/Greedy/shortest-job-first-or-sjf-cpu-scheduling>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\ShortestJobFirst.java

1. Insert Interval :

<https://www.youtube.com/watch?si=a7aPuIw16zDx2lAa&v=xxRE-46OCC8&feature=youtu.be>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\InsertIntervals.java

1. Merge overlapping Interval :

<https://takeuforward.org/data-structure/merge-overlapping-sub-intervals/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\MergeIntervals.java

1. Non – Overlapping Intervals :

<https://www.youtube.com/watch?v=HDHQ8lAWakY>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\Blind75\_byStrivers\src\Greedy\_Algorithm\NonOverlappingIntervals.java

**Sorting Algorithm:**

1. Selection Sort :

<https://takeuforward.org/sorting/selection-sort-algorithm/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\sorting\_algorithms\SelectionSortImpl.java

1. Bubble Sort :

<https://takeuforward.org/data-structure/bubble-sort-algorithm/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\sorting\_algorithms\BubbleSortImpl.java

1. Insertion Sort :

<https://takeuforward.org/data-structure/insertion-sort-algorithm/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\sorting\_algorithms\InsertionSortImpl.java

1. Merge Sort:

<https://takeuforward.org/data-structure/merge-sort-algorithm/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\sorting\_algorithms\MergeSortImpl.java

1. Quick Sort:

<https://takeuforward.org/data-structure/quick-sort-algorithm/>

F:\eclipse\_workspaces\workspace\_DataStrucres\_Algo\_udemy\CodingExamples\_TCS\_NQT\_sheet\src\sorting\_algorithms\QuickSortImpl.java