**30 days of DSA Challenge**

* ***”prepare for placement interviews”***

# **Day-1**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Set Matrix Zeros** | [Solution](https://takeuforward.org/data-structure/set-matrix-zero/) | [Click](https://leetcode.com/problems/set-matrix-zeroes/) | [Youtube](https://www.youtube.com/watch?v=M65xBewcqcI&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=7) | [Code](https://github.com/striver79/SDESheet/blob/main/setMatrixZerosCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/setMatrixZerosJava) |
| 2 | **Pascal Triangle** | [Solution](https://takeuforward.org/data-structure/program-to-generate-pascals-triangle/) | [Click](https://leetcode.com/problems/pascals-triangle/) | [YouTube](https://www.youtube.com/watch?v=6FLvhQjZqvM&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=8) | [Code](https://github.com/striver79/SDESheet/blob/main/pascalTriangleCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/pascalTriangleJava) |
| 3. | **Next Permutation** | [Solution](https://takeuforward.org/data-structure/next_permutation-find-next-lexicographically-greater-permutation/) | [Click](https://leetcode.com/problems/next-permutation/) | [YouTube](https://www.youtube.com/watch?v=LuLCLgMElus&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=9) | [Code](https://github.com/striver79/SDESheet/blob/main/nextPermutationC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/nextPermutationJava) |
| 4. | **Kadane’s Algorithm** | [Solution](https://takeuforward.org/data-structure/kadanes-algorithm-maximum-subarray-sum-in-an-array/) | [Click](https://leetcode.com/problems/maximum-subarray/) | [YouTube](https://www.youtube.com/watch?v=w_KEocd__20&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=5) | [Code](https://github.com/striver79/SDESheet/blob/main/KadaneCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/KadaneJava) |
| 5. | **Sort an array of 0’s 1’s 2’s** | [Solution](https://takeuforward.org/data-structure/sort-an-array-of-0s-1s-and-2s/) | [Click](https://leetcode.com/problems/sort-colors/) | [YouTube](https://www.youtube.com/watch?v=oaVa-9wmpns&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=2) | [Code](https://github.com/striver79/SDESheet/blob/main/sortColorsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/sortColorsJava) |
| 6. | **Stock Buy and Sell** | [Solution](https://takeuforward.org/data-structure/stock-buy-and-sell/) | [Click](https://leetcode.com/problems/best-time-to-buy-and-sell-stock/) | [YouTube](https://www.youtube.com/watch?v=eMSfBgbiEjk&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=11) | [Code](https://github.com/striver79/SDESheet/blob/main/stockBuyAndSellCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/stockBuyAndSellJava) |

# **Day-2**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Rotate Matrix** | [Solution](https://takeuforward.org/data-structure/rotate-image-by-90-degree/) | [Click](https://leetcode.com/problems/rotate-image/) | [Youtube](https://www.youtube.com/watch?v=Y72QeX0Efxw&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=12) | [Code](https://github.com/striver79/SDESheet/blob/main/rotateImageCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/rotateImageJava) |
| 2 | **Merge Overlapping Subintervals** | [Solution](https://takeuforward.org/data-structure/merge-overlapping-sub-intervals/) | [Click](https://leetcode.com/problems/merge-intervals/) | [YouTube](https://www.youtube.com/watch?v=2JzRBPFYbKE&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=6) | [Code](https://github.com/striver79/SDESheet/blob/main/mergeIntervalsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/mergeIntervalsJava) |
| 3. | **Merge two sorted Arrays without** **extra space** | [Solution](https://takeuforward.org/data-structure/merge-two-sorted-arrays-without-extra-space/) | [Click](https://leetcode.com/problems/merge-sorted-array/) | [YouTube](https://www.youtube.com/watch?v=hVl2b3bLzBw&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=4) | [Code](https://github.com/striver79/SDESheet/blob/main/MergeTwoSortedArraysCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/MergeTwoSortedArraysJava) |
| 4. | **Find the duplicate in an array** **of N+1 integers.** | Coming Soon | [Click](https://leetcode.com/problems/find-the-duplicate-number/) | [YouTube](https://www.youtube.com/watch?v=32Ll35mhWg0&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=1) | [Code](https://github.com/striver79/SDESheet/blob/main/findDuplicatesCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/findDuplicatesJava) |
| 5. | **Repeat and Missing Number** | [Solution](https://takeuforward.org/data-structure/find-the-repeating-and-missing-numbers/) | [Click](https://www.interviewbit.com/problems/repeat-and-missing-number-array/) | [YouTube](https://www.youtube.com/watch?v=5nMGY4VUoRY&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=3) | [Code](https://github.com/striver79/SDESheet/blob/main/missingAndRepeatingNumberCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/missingAndRepeatingNumberJava) |
| 6. | **Inversion of Array** (Pre-req: Merge Sort) | [Solution](https://takeuforward.org/data-structure/count-inversions-in-an-array/) | [Click](https://www.codingninjas.com/codestudio/problems/count-inversions_615) | [YouTube](https://www.youtube.com/watch?v=kQ1mJlwW-c0) | Code | Code |

# **Day-3**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Search in a 2d Matrix** | [Solution](https://takeuforward.org/data-structure/search-in-a-sorted-2d-matrix/) | [Click](https://leetcode.com/problems/search-a-2d-matrix/) | [Youtube](https://www.youtube.com/watch?v=ZYpYur0znng&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=14) | [Code](https://github.com/striver79/SDESheet/blob/main/searchInA2DMatrix) | [Code](https://github.com/striver79/SDESheet/blob/main/searchInA2DMatrixJava) |
| 2 | **Pow(X,n)** | [Solution](https://takeuforward.org/data-structure/implement-powxn-x-raised-to-the-power-n/) | [Click](https://leetcode.com/problems/powx-n/) | [YouTube](https://www.youtube.com/watch?v=l0YC3876qxg&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=15) | [Code](https://github.com/striver79/SDESheet/blob/main/powCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/powJava) |
| 3. | **Majority Element (>N/2 times)** | [Solution](https://takeuforward.org/data-structure/find-the-majority-element-that-occurs-more-than-n-2-times/) | [Click](https://leetcode.com/problems/majority-element/) | [YouTube](https://www.youtube.com/watch?v=AoX3BPWNnoE&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=16) | [Code](https://github.com/striver79/SDESheet/blob/main/majorityElement-1Cpp) | [Code](https://github.com/striver79/SDESheet/blob/main/majorityElement-1Java) |
| 4. | **Majority Element (>N/3 times)** | [Solution](https://takeuforward.org/data-structure/majority-elementsn-3-times-find-the-elements-that-appears-more-than-n-3-times-in-the-array/) | [Click](https://leetcode.com/problems/majority-element-ii/) | [YouTube](https://www.youtube.com/watch?v=yDbkQd9t2ig&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=17) | [Code](https://github.com/striver79/SDESheet/blob/main/majorityElement-2Cpp) | [Code](https://github.com/striver79/SDESheet/blob/main/majorityElement-2Java) |
| 5. | **Grid Unique Paths** | [Solution](https://takeuforward.org/data-structure/grid-unique-paths-count-paths-from-left-top-to-the-right-bottom-of-a-matrix/) | [Click](https://leetcode.com/problems/unique-paths/) | [YouTube](https://www.youtube.com/watch?v=t_f0nwwdg5o&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=18) | [Code](https://github.com/striver79/SDESheet/blob/main/GridUniquePathsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/GridUniquePathsJava) |
| 6. | **Reverse Pairs (Leetcode)** | [Solution](https://takeuforward.org/data-structure/count-reverse-pairs/) | [Click](https://leetcode.com/problems/reverse-pairs/) | [YouTube](https://www.youtube.com/watch?v=S6rsAlj_iB4&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=19) | [Code](https://github.com/striver79/SDESheet/blob/main/reversePairsJava) | [Code](https://github.com/striver79/SDESheet/blob/main/reversePairsCpp) |

# **Day-4**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **2-Sum-Problem** | [Solution](https://takeuforward.org/data-structure/two-sum-check-if-a-pair-with-given-sum-exists-in-array/) | [Click](https://leetcode.com/problems/two-sum/) | [Youtube](https://www.youtube.com/watch?v=dRUpbt8vHpo&list=PLgUwDviBIf0rVwua0kKYlsS_ik_1lyVK_&index=2) | [Code](https://github.com/striver79/SDESheet/blob/main/2SumCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/2SumJava) |
| 2 | **4-sum-Problem** | [Solution](https://takeuforward.org/data-structure/4-sum-find-quads-that-add-up-to-a-target-value/) | [Click](https://leetcode.com/problems/4sum/) | [YouTube](https://www.youtube.com/watch?v=4ggF3tXIAp0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=21) | [Code](https://github.com/striver79/SDESheet/blob/main/4SumCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/4SumJava) |
| 3. | **Longest Consecutive Sequence** | Coming Soon | [Click](https://leetcode.com/problems/longest-consecutive-sequence/) | [YouTube](https://www.youtube.com/watch?v=qgizvmgeyUM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=22) | [Code](https://github.com/striver79/SDESheet/blob/main/longestConsSequenceCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/longestConsSequenceJava) |
| 4. | **Largest Subarray with 0 sum** | [Solution](https://takeuforward.org/data-structure/length-of-the-longest-subarray-with-zero-sum/) | [Click](https://practice.geeksforgeeks.org/problems/largest-subarray-with-0-sum/1) | [YouTube](https://www.youtube.com/watch?v=xmguZ6GbatA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=23) | [Code](https://github.com/striver79/SDESheet/blob/main/largestSubarrayWith0SumCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/largestSubarrayWith0SumJava) |
| 5. | **Count number of subarrays with** **given Xor** K | [Solution](https://takeuforward.org/data-structure/count-the-number-of-subarrays-with-given-xor-k/) | [Click](https://www.interviewbit.com/problems/subarray-with-given-xor/) | [YouTube](https://www.youtube.com/watch?v=lO9R5CaGRPY&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=25) | [Code](https://github.com/striver79/SDESheet/blob/main/subarrayWithXorKCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/subarrayWithXorK) |
| 6. | **Longest Substring without repeat** | [Solution](https://takeuforward.org/data-structure/length-of-longest-substring-without-any-repeating-character/) | [Click](https://leetcode.com/problems/longest-substring-without-repeating-characters/) | [YouTube](https://www.youtube.com/watch?v=qtVh-XEpsJo&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=26) | [Code](https://github.com/striver79/SDESheet/blob/main/longestSubstringWithoutRepeatCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/longestSubstringWithoutRepeatJava) |

# **Day-5**: Linked List

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Reverse a LinkedList** | [Solution](https://takeuforward.org/data-structure/reverse-a-linked-list/) | [Click](https://leetcode.com/problems/reverse-linked-list/) | [Youtube](https://www.youtube.com/watch?v=iRtLEoL-r-g&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=27) | [Code](https://github.com/striver79/SDESheet/blob/main/reverseALLCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/reverseLLJava) |
| 2 | **Find the middle of LinkedList** | [Solution](https://takeuforward.org/data-structure/find-middle-element-in-a-linked-list/) | [Click](https://leetcode.com/problems/middle-of-the-linked-list/) | [YouTube](https://www.youtube.com/watch?v=sGdwSH8RK-o&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=28) | [Code](https://github.com/striver79/SDESheet/blob/main/middleLLCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/middleLLJava) |
| 3. | **Merge two sorted Linked List** (use method used in mergeSort) | [Solution](https://takeuforward.org/data-structure/merge-two-sorted-linked-lists/) | [Click](https://leetcode.com/problems/merge-two-sorted-lists/) | [YouTube](https://www.youtube.com/watch?v=Xb4slcp1U38&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=29) | [Code](https://github.com/striver79/SDESheet/blob/main/mergeTwoListsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/mergeTwoListsCpp) |
| 4. | **Remove N-th node from back of** **LinkedList** | [Solution](https://takeuforward.org/data-structure/remove-n-th-node-from-the-end-of-a-linked-list/) | [Click](https://leetcode.com/problems/remove-nth-node-from-end-of-list/) | [YouTube](https://www.youtube.com/watch?v=Lhu3MsXZy-Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=30) | [Code](https://github.com/striver79/SDESheet/blob/main/removeNthNodeEndCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/removeNthNodeEndJava) |
| 5. | **Add two numbers as LinkedList** | [Solution](https://takeuforward.org/data-structure/add-two-numbers-represented-as-linked-lists/) | [Click](https://leetcode.com/problems/add-two-numbers/) | [YouTube](https://www.youtube.com/watch?v=LBVsXSMOIk4&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=32) | [Code](https://github.com/striver79/SDESheet/blob/main/add2LLCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/add2LLJava) |
| 6. | **Delete a given Node when a node** **is given.** **(0(1) solution)** | Coming Soon | [Click](https://leetcode.com/problems/delete-node-in-a-linked-list/) | [YouTube](https://www.youtube.com/watch?v=icnp4FJdZ_c&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=31) | [Code](https://github.com/striver79/SDESheet/blob/main/deleteANodeCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/deleteANodeJava) |

# **Day-6**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Find intersection point** **of Y LinkedList** | [Click](https://leetcode.com/problems/intersection-of-two-linked-lists/) | [Youtube](https://www.youtube.com/watch?v=u4FWXfgS8jw&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=33) | [Code](https://github.com/striver79/SDESheet/blob/main/intersectionYLLCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/intersectionYLLJava) |
| 2 | **Detect a cycle in Linked List** | [Click](https://leetcode.com/problems/linked-list-cycle/) | [YouTube](https://www.youtube.com/watch?v=354J83hX7RI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=34) | [Code](https://github.com/striver79/SDESheet/blob/main/detectCycleCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/detectCycleJava) |
| 3. | **Reverse a LinkedList in** **groups of size k.** | [Click](https://leetcode.com/problems/reverse-nodes-in-k-group/) | [YouTube](https://www.youtube.com/watch?v=Of0HPkk3JgI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=34) | [Code](https://github.com/striver79/SDESheet/blob/main/reverseLLSizeKCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/reverseLLSizeKJava) |
| 4. | **Check if a LinkedList is** **palindrome or not.** | [Click](https://leetcode.com/problems/palindrome-linked-list/) | [YouTube](https://www.youtube.com/watch?v=-DtNInqFUXs&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=36) | [Code](https://github.com/striver79/SDESheet/blob/main/checkForPalindromeLLJava) | [Code](https://github.com/striver79/SDESheet/blob/main/checkForPalindromeLLJava) |
| 5. | **Find the starting point** **of the Loop of LinkedList** | [Click](https://leetcode.com/problems/linked-list-cycle-ii/) | [YouTube](https://www.youtube.com/watch?v=QfbOhn0WZ88&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=37) | [Code](https://github.com/striver79/SDESheet/blob/main/startingPointInLoopInLLCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/startingPointInLoopInLLJava) |
| 6. | **Flattening of a LinkedList** | [Click](https://practice.geeksforgeeks.org/problems/flattening-a-linked-list/1) | [YouTube](https://www.youtube.com/watch?v=ysytSSXpAI0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=38) | [Code](https://github.com/striver79/SDESheet/blob/main/flatteningOfLinkedListCPP) | [Code](https://github.com/striver79/SDESheet/blob/main/flatteningOfLinkedListJava) |
| 7. | **Rotate a LinkedList** | [Click](https://leetcode.com/problems/rotate-list/description/) | [YouTube](https://www.youtube.com/watch?v=9VPm6nEbVPA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=39) | [Code](https://github.com/striver79/SDESheet/blob/main/rotateListCPP) | [Code](https://github.com/striver79/SDESheet/blob/main/rotateListJava) |

# **Day-7**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Clone a Linked List with** **random and next pointer** |  | [Click](https://leetcode.com/problems/copy-list-with-random-pointer/) | [Youtube](https://www.youtube.com/watch?v=VNf6VynfpdM) | [Code](https://github.com/striver79/SDESheet/blob/main/setMatrixZerosCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/setMatrixZerosJava) |
| 2 | **3 sum** | [Solution](https://takeuforward.org/data-structure/3-sum-find-triplets-that-add-up-to-a-zero/) | [Click](https://leetcode.com/problems/3sum/) | [YouTube](https://www.youtube.com/watch?v=onLoX6Nhvmg&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=41) | [Code](https://github.com/striver79/SDESheet/blob/main/3sumC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/3sumJava) |
| 3. | **Trapping rainwater** | [Solution](https://takeuforward.org/data-structure/trapping-rainwater/) | [Click](https://leetcode.com/problems/trapping-rain-water/) | [YouTube](https://www.youtube.com/watch?v=m18Hntz4go8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=42) | [Code](https://github.com/striver79/SDESheet/blob/main/trappingRainWaterC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/trappingRainWaterJava) |
| 4. | **Remove Duplicate from Sorted array** | [Solution](https://takeuforward.org/data-structure/remove-duplicates-in-place-from-sorted-array/) | [Click](https://leetcode.com/problems/remove-duplicates-from-sorted-array/) | [YouTube](https://www.youtube.com/watch?v=Fm_p9lJ4Z_8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=43) | [Code](https://github.com/striver79/SDESheet/blob/main/removeDuplicatesC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/removeDuplicatesJava) |
| 5. | **Max consecutive ones** | [Solution](https://takeuforward.org/data-structure/count-maximum-consecutive-ones-in-the-array/) | [Click](https://leetcode.com/problems/max-consecutive-ones/) | [YouTube](https://www.youtube.com/watch?v=Mo33MjjMlyA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=44) | [Code](https://github.com/striver79/SDESheet/blob/main/consecutiveOnesCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/consecutiveOnesJava) |

# **Day-8**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **N meeting in one room** |  | [Click](https://practice.geeksforgeeks.org/problems/n-meetings-in-one-room-1587115620/1) | [Youtube](https://www.youtube.com/watch?v=II6ziNnub1Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=45) | [Code](https://github.com/striver79/SDESheet/blob/main/NmeetingsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/NmeetingsJava) |
| 2 | **Minimum number of platforms** **required for a railway** | [Solution](https://takeuforward.org/data-structure/minimum-number-of-platforms-required-for-a-railway/) | [Click](https://practice.geeksforgeeks.org/problems/minimum-platforms-1587115620/1) | [YouTube](https://www.youtube.com/watch?v=dxVcMDI7vyI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=46) | [Code](https://github.com/striver79/SDESheet/blob/main/minimumPlatformsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/minimumPlatformsJava) |
| 3. | **Job sequencing Problem** |  | [Click](https://practice.geeksforgeeks.org/problems/job-sequencing-problem-1587115620/1) | [YouTube](https://www.youtube.com/watch?v=LjPx4wQaRIs&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=47) | [Code](https://github.com/striver79/SDESheet/blob/main/jobSequencingCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/jobSequencingJava) |
| 4. | **Fractional Knapsack Problem** | [Solution](https://takeuforward.org/data-structure/fractional-knapsack-problem-greedy-approach/) | [Click](https://practice.geeksforgeeks.org/problems/fractional-knapsack-1587115620/1) | [YouTube](https://www.youtube.com/watch?v=F_DDzYnxO14&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=49) | [Code](https://github.com/striver79/SDESheet/blob/main/fractionalKnapsackC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/fractionalKnapsackJava) |
| 5. | **Greedy algorithm to find** **minimum number of coins** | [Solution](https://takeuforward.org/data-structure/find-minimum-number-of-coins/) | [Click](https://www.geeksforgeeks.org/find-minimum-number-of-coins-that-make-a-change/) | [YouTube](https://www.youtube.com/watch?v=mVg9CfJvayM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=48) | Code | Code |
| 6. | **Activity Selection** **(it is the same as N meeting in one room)** | [Solution](https://takeuforward.org/data-structure/n-meetings-in-one-room/) | [Click](https://practice.geeksforgeeks.org/problems/n-meetings-in-one-room-1587115620/1) | [Youtube](https://www.youtube.com/watch?v=II6ziNnub1Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=45) | [Code](https://github.com/striver79/SDESheet/blob/main/NmeetingsCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/NmeetingsJava) |

# **Day-9**: **Recursion**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Subset Sums** | [Solution](https://takeuforward.org/data-structure/subset-sum-sum-of-all-subsets/) | [Click](https://practice.geeksforgeeks.org/problems/subset-sums2234/1) | [Youtube](https://www.youtube.com/watch?v=rYkfBRtMJr8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=52) | [Code](https://github.com/striver79/SDESheet/blob/main/SubsetSumsC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/SubsetSumsJava) |
| 2 | **Subset-II** | [Solution](https://takeuforward.org/data-structure/subset-ii-print-all-the-unique-subsets/) | [Click](https://leetcode.com/problems/subsets-ii/) | [YouTube](https://www.youtube.com/watch?v=RIn3gOkbhQE&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=53) | [Code](https://github.com/striver79/SDESheet/blob/main/Subset-II_Cpp) | [Code](https://github.com/striver79/SDESheet/blob/main/Subset-II_Java) |
| 3. | **Combination sum-1** | [Solution](https://takeuforward.org/data-structure/combination-sum-1/) | [Click](https://leetcode.com/problems/combination-sum/) | [YouTube](https://www.youtube.com/watch?v=OyZFFqQtu98&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=49) | [Code](https://github.com/striver79/SDESheet/blob/main/combinationSumC%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/combinationSumJava) |
| 4. | **Combination sum-2** |  | [Click](https://leetcode.com/problems/combination-sum-ii/) | [YouTube](https://www.youtube.com/watch?v=G1fRTGRxXU8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=50) | [Code](https://github.com/striver79/SDESheet/blob/main/combinationSum2C%2B%2B) | [Code](https://github.com/striver79/SDESheet/blob/main/combinationSum2Java) |
| 5. | **Palindrome Partitioning** | [Solution](https://takeuforward.org/data-structure/palindrome-partitioning/) | [Click](https://leetcode.com/problems/palindrome-partitioning/) | [YouTube](https://www.youtube.com/watch?v=WBgsABoClE0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=51) | [Code](https://github.com/striver79/SDESheet/blob/main/PalindromePartitioningCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/PalindromePartitioningJava) |
| 6. | **K-th permutation Sequence** | [Solution](https://takeuforward.org/data-structure/find-the-duplicate-in-an-array-of-n1-integers/) | [Click](https://leetcode.com/problems/permutation-sequence/) | [YouTube](https://www.youtube.com/watch?v=wT7gcXLYoao&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=55) | [Code](https://github.com/striver79/SDESheet/blob/main/permutationSequenceCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/permutationSequenceJava) |

# **Day-10** : **Recursion & Backtracking**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Print all permutations of a string/array** | [Solution](https://takeuforward.org/data-structure/print-all-permutations-of-a-string-array/) | [Click](https://leetcode.com/problems/permutations/) | [Youtube](https://www.youtube.com/watch?v=f2ic2Rsc9pU&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=52) | Code | Code |
| 2 | **N queens Problem** | [Solution](https://takeuforward.org/data-structure/n-queen-problem-return-all-distinct-solutions-to-the-n-queens-puzzle/) | [Click](https://leetcode.com/problems/n-queens/) | [Youtube](https://www.youtube.com/watch?v=i05Ju7AftcM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=57) | [Code](https://github.com/striver79/SDESheet/blob/main/NQueenEfficient) | [Code](https://github.com/striver79/SDESheet/blob/main/NQueenEfficientJava) |
| 3. | **Sudoku Solver** |  | [Click](https://leetcode.com/problems/sudoku-solver/) | [Youtube](https://www.youtube.com/watch?v=FWAIf_EVUKE&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=58) | [Code](https://github.com/striver79/SDESheet/blob/main/sudokoSolverCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/sudokoSolverJava) |
| 4. | **M coloring Problem** | [Solution](https://takeuforward.org/data-structure/m-coloring-problem/) | [Click](https://practice.geeksforgeeks.org/problems/m-coloring-problem-1587115620/1) | [Youtube](https://www.youtube.com/watch?v=wuVwUK25Rfc&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=59) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/mcoloringgraphCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/mcoloringgraphJava) |
| 5. | **Rat in a Maze** | [Solution](https://takeuforward.org/data-structure/rat-in-a-maze/) | [Click](https://practice.geeksforgeeks.org/problems/rat-in-a-maze-problem/1) | [Youtube](https://www.youtube.com/watch?v=bLGZhJlt4y0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=60) | [Code](https://github.com/striver79/SDESheet/blob/main/ratInAMazeCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/ratInAMazeJava) |
| 6. | **Word Break (print all ways)** |  | Click | Youtube | Code | Code |

# **Day-1**1: **Binary Search**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **The N-th root of an integer** | [Solution](https://takeuforward.org/data-structure/nth-root-of-a-number-using-binary-search/) | Click | [Youtube](https://www.youtube.com/watch?v=WjpswYrS2nY&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=62) | [Code](https://github.com/striver79/SDESheet/blob/main/NthRootOfNumberCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/NthRootOfNumberJava) |
| 2 | **Matrix Median** |  | [Click](https://www.interviewbit.com/problems/matrix-median/) | [Youtube](https://www.youtube.com/watch?v=63fPPOdIr2c&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=62) | [Code](https://github.com/striver79/SDESheet/blob/main/matrixMedianCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/matrixMedianJava) |
| 3. | **Find the element that appears once in a sorted array, and the rest element appears twice (Binary search)** | [Solution](https://takeuforward.org/data-structure/search-single-element-in-a-sorted-array/) | [Click](https://leetcode.com/problems/single-element-in-a-sorted-array/) | [Youtube](https://www.youtube.com/watch?v=PzszoiY5XMQ&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=64) | [Code](https://github.com/striver79/SDESheet/blob/main/SingleElementSortedArrayCPP) | [Code](https://github.com/striver79/SDESheet/blob/main/SingleElementSortedArrayJava) |
| 4. | **Search element in a sorted and rotated array/ find pivot where it is rotated** | [Solution](https://takeuforward.org/data-structure/search-element-in-a-rotated-sorted-array/) | [Click](https://leetcode.com/problems/search-in-rotated-sorted-array/) | [Youtube](https://www.youtube.com/watch?v=r3pMQ8-Ad5s&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=64) | [Code](https://github.com/striver79/SDESheet/blob/main/searchInPivotedArrayCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/searchInPivotedArrayJava) |
| 5. | **Median of 2 sorted arrays** | [Solution](https://takeuforward.org/data-structure/median-of-two-sorted-arrays-of-different-sizes/) | [Click](https://leetcode.com/problems/median-of-two-sorted-arrays/) | [Youtube](https://www.youtube.com/watch?v=NTop3VTjmxk&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=65) | [Code](https://github.com/striver79/SDESheet/blob/main/medianOf2SortedArraysCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/medianOf2SortedArraysJava) |
| 6. | **K-th element of two sorted arrays** | [Solution](https://takeuforward.org/data-structure/k-th-element-of-two-sorted-arrays/) | [Click](https://practice.geeksforgeeks.org/problems/k-th-element-of-two-sorted-array1317/1) | [Youtube](https://www.youtube.com/watch?v=nv7F4PiLUzo&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=66) | [Code](https://github.com/striver79/SDESheet/blob/main/kThLargestElementCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/kThLargestElementJava) |
| 7. | **Allocate Minimum Number of Pages** | [Solution](https://takeuforward.org/data-structure/allocate-minimum-number-of-pages/) | [Click](https://www.interviewbit.com/problems/allocate-books/) | [Youtube](https://www.youtube.com/watch?v=gYmWHvRHu-s&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=69) | [Code](https://github.com/striver79/SDESheet/blob/main/allocateBooksCpp) | [Code](https://github.com/striver79/SDESheet/blob/main/allocateBooksJava) |
| 8. | **Aggressive Cows** | [Solution](https://takeuforward.org/data-structure/aggressive-cows-detailed-solution/) | [Click](https://www.spoj.com/problems/AGGRCOW/) | [Youtube](https://www.youtube.com/watch?v=wSOfYesTBRk&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=70) | [Code](https://github.com/striver79/SDESheet/blob/main/aggressiveCows) | Code |

# **Day-12**: **TRIES (Can be done at Last, but still a very very important topic) Watch this playlist ->**[Link](https://www.youtube.com/watch?v=dBGUmUQhjaM&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp)

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Implement Trie (Prefix Tree)** | [Click](https://leetcode.com/problems/implement-trie-prefix-tree/) | [Youtube](https://www.youtube.com/watch?v=dBGUmUQhjaM&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L1_ImplementTrieCpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L1_ImplementTrieJava) |
| 2 | **Implement Trie – 2 (Prefix Tree)** | [Click](https://bit.ly/3qwT4OL) | [Youtube](https://www.youtube.com/watch?v=K5pcpkEMCN0&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp&index=2) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L2Cpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L2_ImplementTrie2Java) |
| 3. | **Longest String with All Prefixes** | [Click](https://bit.ly/3n3kedU) | [Youtube](https://www.youtube.com/watch?v=AWnBa91lThI&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp&index=3) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L3_CompleteString_Cpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L3_Complete_String_Java) |
| 4. | **Number of Distinct Substrings in a String** | [Click](https://bit.ly/3ocRQW0) | [Youtube](https://www.youtube.com/watch?v=RV0QeTyHZxo&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp&index=4) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L4_DistinctSubstrings_Cpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L4_DistinctSubstrings_Java) |
| 4. | **Power Set (this is very important)** | [Click](https://practice.geeksforgeeks.org/problems/power-set4302/1) | [Youtube](https://www.youtube.com/watch?v=b7AYbpM5YrE&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=67) | [Code](https://github.com/striver79/SDESheet/blob/main/powerSetCpp) | Code |
| 5. | **Maximum XOR of two numbers in an array** | [Click](https://leetcode.com/problems/maximum-xor-of-two-numbers-in-an-array/) | [Youtube](https://www.youtube.com/watch?v=EIhAwfHubE8&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp&index=6) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L5MaxPairCpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L5MaxNumTrieJava) |
| 6. | **Maximum XOR With an Element From Array** | [Click](https://leetcode.com/problems/maximum-xor-with-an-element-from-array/) | [Youtube](https://www.youtube.com/watch?v=Q8LhG9Pi5KM&list=PLgUwDviBIf0pcIDCZnxhv0LkHf5KzG9zp&index=7) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L6MaxXorQueriesCpp) | [Code](https://github.com/striver79/StriversTrieSeries/blob/main/L6MaxXorQueriesJava) |

# **Day-13** : **(Stack and Queue)**

| **Q.NO** | **PROBLEM NAME** | **SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Implement Stack Using Arrays** | [Solution](https://takeuforward.org/data-structure/implement-stack-using-array/) | Click | [Youtube](https://www.youtube.com/watch?v=GYptUgnIM_I&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=68) | Code | Code |
| 2 | **Implement Queue Using Arrays** | [Solution](https://takeuforward.org/data-structure/implement-queue-using-array/) | Click | [Youtube](https://www.youtube.com/watch?v=M6GnoUDpqEE&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=72) | Code | Code |
| 3. | **Implement Stack using Queue (using single queue)** | [Solution](https://takeuforward.org/data-structure/implement-stack-using-single-queue/) | [Click](https://leetcode.com/problems/implement-stack-using-queues/) | [Youtube](https://www.youtube.com/watch?v=jDZQKzEtbYQ&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=74) | Code | Code |
| 4. | **Implement Queue using Stack (0(1) amortized method)** | [Solution](https://takeuforward.org/data-structure/implement-queue-using-stack/) | [Click](https://leetcode.com/problems/implement-queue-using-stacks/) | [Youtube](https://www.youtube.com/watch?v=3Et9MrMc02A&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=75) | Code | Code |
| 5. | **Check for balanced parentheses** | [Solution](https://takeuforward.org/data-structure/check-for-balanced-parentheses/) | [Click](https://leetcode.com/problems/valid-parentheses/) | [Youtube](https://www.youtube.com/watch?v=wkDfsKijrZ8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=74) | Code | Code |
| 6. | **Next Greater Element** | [Solution](https://takeuforward.org/data-structure/next-greater-element-using-stack/) | [Click](https://leetcode.com/problems/next-greater-element-i/) | [Youtube](https://www.youtube.com/watch?v=Du881K7Jtk8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=75) | Code | Code |
| 7. | **Sort a Stack** |  | Click | Youtube | Code | Code |

# **Day-1**4 :

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Next Smaller Element** |  | [Click](https://www.interviewbit.com/problems/nearest-smaller-element/) | Youtube | Code | Code |
| 2 | **LRU cache (IMPORTANT)** | [Solution](https://takeuforward.org/data-structure/implement-lru-cache/) | [Click](https://leetcode.com/problems/lru-cache/) | [Youtube](https://www.youtube.com/watch?v=xDEuM5qa0zg&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=77) | Code | Code |
| 3. | **LFU Cache** |  | [Click](https://leetcode.com/problems/lfu-cache/) | [Youtube](https://www.youtube.com/watch?v=0PSB9y8ehbk&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=79) | Code | Code |
| 4. | **Largest rectangle in a histogram** | [Solution](https://takeuforward.org/data-structure/area-of-largest-rectangle-in-histogram/) | [Click](https://leetcode.com/problems/largest-rectangle-in-histogram/) | [Two-Pass: Youtube](https://www.youtube.com/watch?v=X0X6G-eWgQ8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=81)  [One Pass: Youtube](https://www.youtube.com/watch?v=jC_cWLy7jSI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=82) | Code | Code |
| 5. | **Sliding Window maximum** | [Solution](https://takeuforward.org/data-structure/sliding-window-maximum/) | [Click](https://leetcode.com/problems/sliding-window-maximum/) | [Youtube](https://www.youtube.com/watch?v=CZQGRp93K4k&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=83) | Code | Code |
| 6. | **Implement Min Stack** | [Solution](https://takeuforward.org/data-structure/implement-min-stack-o2n-and-on-space-complexity/) | [Click](https://leetcode.com/problems/min-stack/) | Youtube | Code | Code |
| 7. | **Rotten Orange (Using BFS**) | [Solution](https://takeuforward.org/data-structure/rotten-oranges-min-time-to-rot-all-oranges-bfs/) | [Click](https://leetcode.com/problems/rotting-oranges/) |  |  |  |
| 8. | **Stock Span Problem** |  | [Click](https://leetcode.com/problems/online-stock-span/) |  |  |  |
| 9. | **Find the maximum of minimums of every window size** |  | Click |  |  |  |
| 10. | **The Celebrity Problem** |  | [Click](https://leetcode.com/accounts/login/?next=/problems/find-the-celebrity/) |  |  |  |

# **Day-15**: **String**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Reverse Words in a String** | [Click](https://leetcode.com/problems/reverse-words-in-a-string/) | Youtube | Code | Code |
| 2 | **Longest Palindrome in a string** | [Click](https://leetcode.com/problems/longest-palindromic-substring/) | Youtube | Code | Code |
| 3. | **Roman Number to Integer and vice versa** | [Click](https://leetcode.com/problems/roman-to-integer/) | Youtube | Code | Code |
| 4. | **Implement ATOI/STRSTR** | [Click](https://leetcode.com/problems/string-to-integer-atoi/) | Youtube | Code | Code |
| 5. | **Longest Common Prefix** | [Click](https://leetcode.com/problems/longest-common-prefix/) | Youtube | Code | Code |
| 6. | **Rabin Karp** | [Click](https://leetcode.com/problems/repeated-string-match/discuss/416144/Rabin-Karp-algorithm-C%2B%2B-implementation) | Youtube | Code | Code |

# **Day-16**: **String [Continued]**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Z-Function** | [Click](https://leetcode.com/problems/implement-strstr/) | Youtube | Code | Code |
| 2 | **KMP algo / LPS(pi) array** | [Click](https://leetcode.com/problems/implement-strstr/) | YouTube | Code | Code |
| 3. | **Minimum characters needed** **to be inserted in the beginning** **to make it palindromic** | [Click](https://www.interviewbit.com/problems/minimum-characters-required-to-make-a-string-palindromic/) | YouTube | Code | Code |
| 4. | **Check for Anagrams** | [Click](https://leetcode.com/problems/valid-anagram/) | YouTube | Code | Code |
| 5. | **Count and Say** | [Click](https://leetcode.com/problems/count-and-say/) | YouTube | Code | Code |
| 6. | **Compare version numbers** | [Click](https://leetcode.com/problems/compare-version-numbers/) | YouTube | Code | Code |

# **Day-17**: **Binary Tree** ([Introduction](https://www.youtube.com/watch?v=OYqYEM1bMK8&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk))

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Inorder Traversal** | [Morris Traversal](https://takeuforward.org/data-structure/morris-inorder-traversal-of-a-binary-tree/)  [Iterative/Recursive](https://takeuforward.org/data-structure/inorder-traversal-of-binary-tree/) | [Click](https://leetcode.com/problems/binary-tree-inorder-traversal/) | [Youtube](https://www.youtube.com/watch?v=Z_NEgBgbRVI&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=7) (Recursive) [Youtube](https://www.youtube.com/watch?v=lxTGsVXjwvM&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=11) (Iterative) [Youtube](https://www.youtube.com/watch?v=80Zug6D1_r4&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=38) (Morris Traversal) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/inorderCpp) (Recursive) [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L10_iterativeTraversalInorderCpp) (Iterative) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/inorderJava) (Recursive) [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L10_iterativeTraversalInorderJava) (Iterative) [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L37_inorderMorrisTraversalJava) (Morris) |
| 2 | **Preorder Traversal** | [Morris Traversal](https://takeuforward.org/data-structure/morris-preorder-traversal-of-a-binary-tree/)  [Solution](https://takeuforward.org/data-structure/preorder-traversal-of-binary-tree/) | [Click](https://leetcode.com/problems/binary-tree-preorder-traversal/) | YouTube [Youtube](https://www.youtube.com/watch?v=80Zug6D1_r4&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=38) (Morris Traversal) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L37_preorderMorrisTraversalCpp) (Morris) | Code |
| 3. | **Postorder Traversal** | [Solution](https://takeuforward.org/data-structure/post-order-traversal-of-binary-tree/) | [Click](https://leetcode.com/problems/binary-tree-postorder-traversal/) | YouTube | Code | Code |
| 4. | **LeftView Of Binary Tree** | [Solution](https://takeuforward.org/data-structure/right-left-view-of-binary-tree/) | [Click](https://practice.geeksforgeeks.org/problems/left-view-of-binary-tree/1) | [YouTube](https://www.youtube.com/watch?v=KV4mRzTjlAk&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=25) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L24_RightViewOfBinaryTreeCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L24_RightViewOfBinaryTreeJava) |
| 5. | **Bottom View of Binary Tree** | [Solution](https://takeuforward.org/data-structure/bottom-view-of-a-binary-tree/) | [Click](https://practice.geeksforgeeks.org/problems/bottom-view-of-binary-tree/1) | [YouTube](https://www.youtube.com/watch?v=0FtVY6I4pB8&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=24) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L23_bottomViewOfBtCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L23_bottomViewOfBtJava) |
| 6. | **Top View of Binary Tree** | [Solution](https://takeuforward.org/data-structure/top-view-of-a-binary-tree/) | [Click](https://practice.geeksforgeeks.org/problems/top-view-of-binary-tree/1) | [YouTube](https://www.youtube.com/watch?v=Et9OCDNvJ78&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=23) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L22_topViewOfBtCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L22_topViewOfBtJava) |
| 7. | **Preorder inorder postorder in a single traversal** | [Solution](https://takeuforward.org/data-structure/preorder-inorder-postorder-traversals-in-one-traversal/) |  |  |  |  |
| 8. | **Vertical order traversal** | [Solution](https://takeuforward.org/data-structure/vertical-order-traversal-of-binary-tree/) |  |  |  |  |
| 9. | **Root to node path in a Binary Tree** | [Solution](https://takeuforward.org/data-structure/print-root-to-node-path-in-a-binary-tree/) |  |  |  |  |
| 10. | **Max width of a Binary Tree** | [Solution](https://takeuforward.org/data-structure/maximum-width-of-a-binary-tree/) |  |  |  |  |

# **Day-18**: **Binary Tree** [Continued]

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Level order Traversal** / **Level order traversal in spiral form** | [Solution](https://takeuforward.org/data-structure/level-order-traversal-of-a-binary-tree/) | [Click](https://leetcode.com/problems/binary-tree-level-order-traversal/) | Youtube | Code | Code |
| 2 | **Height of a Binary Tree** | [Solution](https://takeuforward.org/data-structure/maximum-depth-of-a-binary-tree/) | [Click](https://leetcode.com/problems/maximum-depth-of-binary-tree/) | YouTube | Code | Code |
| 3. | **Diameter of Binary Tree** | [Solution](https://takeuforward.org/data-structure/calculate-the-diameter-of-a-binary-tree/) | [Click](https://leetcode.com/problems/diameter-of-binary-tree/) | YouTube | Code | Code |
| 4. | **Check if the Binary tree is height-balanced or not** | [Solution](https://takeuforward.org/data-structure/check-if-the-binary-tree-is-balanced-binary-tree/) | [Click](https://leetcode.com/problems/balanced-binary-tree/) | YouTube | Code | Code |
| 5. | **LCA in Binary Tree** | [Solution](https://takeuforward.org/data-structure/lowest-common-ancestor-for-two-given-nodes/) | [Click](https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-tree/) | YouTube | Code | Code |
| 6. | **Check if two trees are identical or not** | [Solution](https://takeuforward.org/data-structure/check-if-two-trees-are-identical/) | [Click](https://leetcode.com/problems/same-tree/) | YouTube | Code | Code |
| 7. | **Zig Zag Traversal of Binary Tree** | [Solution](https://takeuforward.org/data-structure/zig-zag-traversal-of-binary-tree/) |  |  |  |  |
| 8. | **Boundary Traversal of Binary Tree** | [Solution](https://takeuforward.org/data-structure/boundary-traversal-of-a-binary-tree/) |  |  |  |  |

# **Day-19**: **Binary Tree** [Continued]

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Maximum path sum** | [Solution](https://takeuforward.org/data-structure/maximum-sum-path-in-binary-tree/) | [Click](https://leetcode.com/problems/binary-tree-maximum-path-sum/) | Youtube | Code | Code |
| 2 | **Construct Binary Tree from inorder and preorder** |  | [Click](https://leetcode.com/problems/construct-binary-tree-from-preorder-and-inorder-traversal/) | YouTube | Code | Code |
| 3. | **Construct Binary Tree from Inorder and Postorder** |  | [Click](https://leetcode.com/problems/construct-binary-tree-from-inorder-and-postorder-traversal/) | YouTube | Code | Code |
| 4. | **Symmetric Binary Tree** | [Solution](https://takeuforward.org/data-structure/check-for-symmetrical-binary-tree/) | [Click](https://leetcode.com/problems/symmetric-tree/) | YouTube | Code | Code |
| 5. | **Flatten Binary Tree to LinkedList** |  | [Click](https://leetcode.com/problems/flatten-binary-tree-to-linked-list/) | YouTube | Code | Code |
| 6. | **Check if Binary Tree is the mirror of itself or not** |  | [Click](https://practice.geeksforgeeks.org/problems/mirror-tree/1) | YouTube | Code | Code |

# **Day-20**: **Binary Search Tree**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Populate Next Right pointers of Tree** | [Click](https://leetcode.com/problems/populating-next-right-pointers-in-each-node/) | Youtube | Code | Code |
| 2 | **Search given Key in BST** | [Click](https://leetcode.com/problems/search-in-a-binary-search-tree/) | YouTube | Code | Code |
| 3. | **Construct BST from given keys** | [Click](https://leetcode.com/problems/convert-sorted-array-to-binary-search-tree/) | YouTube | Code | Code |
| 4. | **Check is a BT is BST or not** | [Click](https://leetcode.com/problems/validate-binary-search-tree/) | YouTube | Code | Code |
| 5. | **Find LCA of two nodes in BST** | [Click](https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-search-tree/) | YouTube | Code | Code |
| 6. | **Find the inorder predecessor/successor of a given Key in BST.** | [Click](https://practice.geeksforgeeks.org/problems/predecessor-and-successor/1) | YouTube | Code | Code |

# **Day-21**: **Binary Search Tree [Continued]**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1. | **Floor in a BST** |  | Click | [Youtube](https://www.youtube.com/watch?v=xm_W1ub-K-w&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=43) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L42_floorInBstCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L42_floorInBstJava) |
| 2. | **Ceil in a BST** |  | Click | [Youtube](https://www.youtube.com/watch?v=KSsk8AhdOZA&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=42) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L41_ceilBstCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L41_ceilBstJava) |
| 3. | **Find K-th smallest element in BST** | [Solution](https://takeuforward.org/data-structure/kth-largest-smallest-element-in-binary-search-tree/) | [Click](https://leetcode.com/problems/kth-smallest-element-in-a-bst/) | YouTube | Code | Code |
| 4. | **Find K-th largest element in BST** | [Solution](https://takeuforward.org/data-structure/kth-largest-smallest-element-in-binary-search-tree/) | [Click](https://practice.geeksforgeeks.org/problems/kth-largest-element-in-bst/1) | YouTube | Code | Code |
| 5. | **Find a pair with a given sum in BST** |  | [Click](https://leetcode.com/problems/two-sum-iv-input-is-a-bst/) | YouTube | Code | Code |
| 6. | **BST iterator** |  | [Click](https://leetcode.com/problems/binary-search-tree-iterator/) | YouTube | Code | Code |
| 7. | **Size of the largest BST in a Binary Tree** |  | [Click](https://leetcode.com/problems/maximum-sum-bst-in-binary-tree/) | YouTube | Code | Code |
| 8. | **Serialize and deserialize Binary Tree** | [Solution](https://takeuforward.org/data-structure/serialize-and-deserialize-a-binary-tree/) | [Click](https://leetcode.com/problems/serialize-and-deserialize-binary-tree/) | [YouTube](https://www.youtube.com/watch?v=-YbXySKJsX8&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=37) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L36_serializeBTCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L36_serializeBTJava) |

# **Day-22**: **Trees [Miscellaneous]**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Binary Tree to Double Linked List** | [Click](https://leetcode.com/problems/flatten-binary-tree-to-linked-list/) | [Youtube](https://www.youtube.com/watch?v=sWf7k1x9XR4&list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk&index=39) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L38_flattenBT2LLCpp) | [Code](https://github.com/striver79/FreeKaTreeSeries/blob/main/L38_flattenBT2LLJava) |
| 2 | **Find median in a stream of running integers.** | [Click](https://leetcode.com/problems/find-median-from-data-stream/) | YouTube | Code | Code |
| 3. | **K-th largest element in a stream.** | [Click](https://leetcode.com/problems/kth-largest-element-in-a-stream/) | YouTube | Code | Code |
| 4. | **Distinct numbers in Window.** | [Click](https://www.interviewbit.com/problems/distinct-numbers-in-window/) | YouTube | Code | Code |
| 5. | **K-th largest element in an unsorted array.** | [Click](https://leetcode.com/problems/kth-largest-element-in-an-array/) | YouTube | Code | Code |
| 6. | **Flood-fill Algorithm** | [Click](https://leetcode.com/problems/flood-fill/) | YouTube | Code | Code |

# **Day-23**: **Graphs – Part 1**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Clone a graph (Not that easy as it looks)** |  | [Click](https://leetcode.com/problems/clone-graph/) | Youtube | Code | Code |
| 2 | **DFS** | [Solution](https://takeuforward.org/data-structure/depth-first-search-dfs-traversal-graph/) | [Click](https://practice.geeksforgeeks.org/problems/depth-first-traversal-for-a-graph/1) | [YouTube](https://www.youtube.com/watch?v=uDWljP2PGmU&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=7) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/DfsCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/DfsJava) |
| 3. | **BFS** | [Solution](https://takeuforward.org/data-structure/breadth-first-searchbfs-level-order-traversal/) | [Click](https://practice.geeksforgeeks.org/problems/bfs-traversal-of-graph/1) | [YouTube](https://www.youtube.com/watch?v=UeE67iCK2lQ&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=6) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/BfsCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/BfsJava) |
| 4. | **Detect A cycle in Undirected Graph using BFS** | [Solution](https://takeuforward.org/data-structure/detect-a-cycle-in-undirected-graph-breadth-first-search/) | [Click](https://leetcode.com/problems/course-schedule/) | [YouTube](https://www.youtube.com/watch?v=A8ko93TyOns&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=8) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckUGBfs) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckUGBfsJava) |
| 5. | **Detect A cycle in Undirected Graph using DFS** | [Solution](https://takeuforward.org/data-structure/cycle-detection-in-undirected-graph-using-dfs/) | [Click](https://leetcode.com/problems/course-schedule/) | [YouTube](https://www.youtube.com/watch?v=Y9NFqI6Pzd4&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=9) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckUGDfs) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckUGDfsJava) |
| 6. | **Detect A cycle in a Directed Graph** **using DFS** | [Solution](https://takeuforward.org/data-structure/detect-a-cycle-in-directed-graph-using-dfs/) | [Click](https://leetcode.com/problems/course-schedule/) | [YouTube](https://www.youtube.com/watch?v=uzVUw90ZFIg&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=12) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckDGDfs) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/cycleCheckDGDfsJava) |
| 7. | **Detect A cycle in a Directed Graph using BFS** |  | [Click](https://leetcode.com/problems/course-schedule/) | [YouTube](https://www.youtube.com/watch?v=V6GxfKDyLBM&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=15) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/checkCycleDGBFSCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/checkCycleDGBFSJava) |
| 8. | **Topological Sort** |  | [Click](https://practice.geeksforgeeks.org/problems/topological-sort/1) | [DFS: YouTube](https://www.youtube.com/watch?v=Yh6EFazXipA&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=13)  [BFS: Youtube](https://www.youtube.com/watch?v=rZv_jHZva34&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=14) | Code | Code |
| 9. | **Number of islands (Do in Grid and Graph both)** |  | [Click](https://leetcode.com/problems/number-of-islands/) | YouTube | Code | Code |
| 10. | **Bipartite Check using BFS** |  | [Click](https://leetcode.com/problems/is-graph-bipartite/) | [YouTube](https://www.youtube.com/watch?v=nbgaEu-pvkU&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=10) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/bipartiteGraphCppBfs) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/bipartiteGraphJavaBfs) |
| 11. | **Bipartite Check using DFS** |  | [Click](https://leetcode.com/problems/is-graph-bipartite/) | [YouTube](https://www.youtube.com/watch?v=uC884ske2uQ&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=11) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/bipartiteGraphCppDfs) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/bipartiteGraphJavaDfs) |

# **Day-24**: **Graphs – Part 2**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Strongly Connected Component(using KosaRaju’s algo)** | [Click](https://leetcode.com/problems/maximum-number-of-non-overlapping-substrings/discuss/766485/kosaraju-algorithm-on) | [Youtube](https://www.youtube.com/watch?v=V8qIqJxCioo&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=27) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/kosarajuCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/kosaRajuJava) |
| 2 | **Dijkstra’s Algorithm** | [Click](https://practice.geeksforgeeks.org/problems/implementing-dijkstra-set-1-adjacency-matrix/1) | [YouTube](https://www.youtube.com/watch?v=jbhuqIASjoM&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=18) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/djisktraCPP) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/djisktraJava) |
| 3. | **Bellman-Ford Algo** | [Click](https://practice.geeksforgeeks.org/problems/distance-from-the-source-bellman-ford-algorithm/0/?fbclid=IwAR2_lL0T84DnciLyzMTQuVTMBOi82nTWNLuXjUgahnrtBgkphKiYk6xcyJU) | [YouTube](https://www.youtube.com/watch?v=75yC1vbS8S8&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=28) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/BellmanFordCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/BellmanFordJava) |
| 4. | **Floyd Warshall Algorithm** | [Click](https://practice.geeksforgeeks.org/problems/implementing-floyd-warshall2042/1) | YouTube | Code | Code |
| 5. | **MST using Prim’s Algo** | [Click](https://practice.geeksforgeeks.org/problems/minimum-spanning-tree/1) | [YouTube](https://www.youtube.com/watch?v=HnD676J56ak&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=20) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/primsAlgoCppEfficient) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/primsAlgoJavaEfficient) |
| 6. | **MST using Kruskal’s Algo** | [Click](https://practice.geeksforgeeks.org/problems/minimum-spanning-tree/1) | [YouTube](https://www.youtube.com/watch?v=1KRmCzBl_mQ&list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw&index=24) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/KruskalAlgorithmCpp) | [Code](https://github.com/striver79/StriversGraphSeries/blob/main/KruskalAlgoJava) |

# **Day-25**: **Dynamic Programming – Part 1**

| **Q.NO** | **PROBLEM NAME** | **DETAILED SOLUTION** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | **Max Product Subarray** |  | [Click](https://leetcode.com/problems/maximum-product-subarray/) | Youtube | Code | Code |
| 2 | **Longest Increasing Subsequence** |  | [Click](https://leetcode.com/problems/longest-increasing-subsequence/) | YouTube | Code | Code |
| 3. | **Longest Common Subsequence** | [Solution](https://takeuforward.org/data-structure/longest-consecutive-sequence-in-an-array/) | [Click](https://leetcode.com/problems/longest-common-subsequence/) | YouTube | Code | Code |
| 4. | **0-1 Knapsack** |  | [Click](https://leetcode.com/problems/ones-and-zeroes/) | YouTube | Code | Code |
| 5. | **Edit Distance** |  | [Click](https://leetcode.com/problems/edit-distance/) | YouTube | Code | Code |
| 6. | **Maximum sum increasing subsequence** |  | [Click](https://practice.geeksforgeeks.org/problems/maximum-sum-increasing-subsequence4749/1) | YouTube | Code | Code |
| 7. | **Matrix Chain Multiplication** |  | [Click](https://practice.geeksforgeeks.org/problems/matrix-chain-multiplication0303/1) | YouTube | Code | Code |

# **Day-26**: **Dynamic Programming – Part 2**

| **Q.NO** | **PROBLEM NAME** | **PROBLEM LINK** | **VIDEO SOLUTION** | **C++ CODE** | **JAVA CODE** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Maximum sum path in the matrix, (count paths and similar type do, also backtrack to find the maximum path)** | [Click](https://leetcode.com/problems/minimum-path-sum/) | Youtube | Code | Code |
| 2 | **Coin change** | [Click](https://leetcode.com/problems/coin-change/) | YouTube | Code | Code |
| 3. | **Subset Sum** | [Click](https://leetcode.com/problems/partition-equal-subset-sum/) | YouTube | Code | Code |
| 4. | **Rod Cutting** | [Click](https://leetcode.com/problems/minimum-cost-to-cut-a-stick/) | YouTube | Code | Code |
| 5. | **Egg Dropping** | [Click](https://practice.geeksforgeeks.org/problems/egg-dropping-puzzle-1587115620/1) | YouTube | Code | Code |
| 6. | **Word Break** | [Click](https://practice.geeksforgeeks.org/problems/word-break1352/1) | YouTube | Code | Code |
| 7. | **Palindrome Partitioning (MCM Variation)** | [Click](https://practice.geeksforgeeks.org/problems/palindromic-patitioning4845/1) | Youtube | Code | Code |
| 8. | **Maximum profit in Job scheduling** | [Click](https://practice.geeksforgeeks.org/problems/job-sequencing-problem-1587115620/1) | Youtube | Code | Code |

# **Day-27:**

1. Revise OS notes that you would have made during your sem
2. If not made notes, spend 2 or 3  days and make notes from Knowledge Gate.

# **Day-28:**

1. Revise DBMS notes that you would have made during your semesters.
2. If not made notes, spend 2 or 3  days and make notes from Knowledge Gate.

# **Day-29:**

1. Revise CN notes, that you would have made during your sem.
2. If not made notes, spend 2 or 3  days and make notes from Knowledge Gate.

# **Day-30:**

1. Make a note of how will your represent your projects, and prepare all questions related to tech which you have used in your projects. Prepare a note which you can say for 3-10 minutes when he asks you that say something about the project.

# Prize:

*Hurrah!! You are ready for your placement after a month of hard work without a cheat day.*

~Striver