Array:

[Easy]

1- Stock max profit. Buy and Sell once.(Minimum size battery needed by robot to complete a hill journey) / buy and sell any number of times

2- You have integer array of N values. Return minimum integer which can’t be represented as the sum of any sub-set of this array.

3- Remove repetitions from a sorted array.

4- Find common elements in 2 sorted array.

5- Find if a pair exist with sum as K in a sorted Array.

6- Majority element of sorted array.

7- Dutch Flag Problem(even-odd)

8- French Flag Problem(3 class)

9- Find majority element in an array

10- Find 1st missing positive number in a array

11- Compute max product of all but one entry. Array can has positive, negative or 0. You can’t use division.

12- Compute longest continuous in creasing sub array.

13- Compute array such that Arr[i] is product of all element except Arr[i] initially. You can’t use division. Extra space as O(1)

14- Apply permutation to an array.

15- Generate next larger permutation of an array.

16- From an integer array of N distinct integers, return a random set of K integers. All k size sunsets should be equally likely. You have a rand(min, max) function / generate a random permutation

17- Check if two rectangle intersects. Rectangles are parallel to X and Y axis.

18- Merge two sorted A and B array. A is large enough to have extra empty space to include B.

19- Sort an array of string so that all anagrams will be together

20- Search in sorted array on unknown length

21- Two integer array given. Find pair having elements from different array with minimum diff.

[Moderate]

151- Rearrange integer array elements as alternate peak and valley i.e. make an array like

A[0] <= A[1] >= A[2] <= A[3] >= A[4] <= A[5] etc

152- Sub-Array with Maximum sum( complexity N3 => N2 => N)

153- Find median of 2 sorted Array.

153A- Same size

153B- Different size

154- Find kth smallest element of an array.

155- Find if an array have majority element.

156- Minimum subarray having all elements or sum > K.

157- Stock max profit. Buy and Sell twice

158- In 2D array all ways to reach from [0][0] to [N-1][M-1] by moving either right or down only

159- Find celebrity

160- <Zero Matrix> Fill entire ith row and jth column with 0’s, if arr[i][j] is 0

161- Rotate a N*N 2D integer array 90* clockwise

162- Search in 2D array sorted by both row and column wise

163- Shortest sub array in a longer array which contains all the elements of a smaller array .

[Tough]

251- Find Largest SubMatrix with maximum sum. in 2D integer array.

252- Find largest subMatrix with all 1’s in a 2D array of 1 & 0 only.

253- Calculate number of subArrays with same number of 1 & 0, in a array having 1 & 0 only

254- Given an array arr[], find the maximum j – i such that arr[j] > arr[i]

String:

[Easy]

301- Check if a String is rotation of other.

302- Rotate a string by ith index.

303- Check is S1 is Substring of S2. If yes, return its staring index of 1st match.

304- Return Nth number of Look and Say sequence ( 1, 11, 21, 1211, 111221, 312211, 13112221, 1113213211 … etc )

305- Implement run length encoding and decoding

306- SnackString of sinusoidal representation of input string.

307- Check if string has all unique characters.

308- Replace all space in a string by ‘%20’. String has enough extra space.

309- Check if one string is permutation of a palindrome.

310- Compress string with continuous count if possible ( aabbbcdeeaaa => a2b3c1d1e2a3 )

*311- Convert spreadsheet(xl) column Id to corresponding no

*312- All mnemonics of a number(print only valid words)

313- Check if one string is permutation of other

[Moderate]

351- Reverse sequence of words in a string.

352- Largest sub-string with matching parentheses

353- Pattern matching. pattern has only two type of chars A & B. like pattern - aaba , string - catcatgocat

354- In a list find largest word which could be build by other words

[Tough]

381- Find substring in O(N + M) time

382- Minimum substring having all words of a given sentence.

Easy - Having all chars

383- Print all possible IP address from a integer string with missing dots(.) like 165871345 could be 165.87.1.137 and could be many more

384- <Re-space>: enter missing spaces in a sentences with minimizing unrecognized chars.

Stack/Queue:

[Easy]

401- Implement Queue by 2 stacks

402- Implement 3 stack using an Array

403- <Validate parentheses - Stack> Check whether the given expression has balanced symbols. such as (, {, or [

404- Prefix and postfix notions are methods of writing mathematical expressions without parenthesis. Generate and Evaluate a postfix and prefix expression [Stack]

405- Implement Stack using 2 queues

406- Find nearest lesser element in the left array for all the elements

407- Find largest spans of a stock(maximum consecutive days for which stock price was less or equal of current day price)

408- implement SetOfStacks when size of one stack is limited.

409-

[Moderate]

451- Implement Min Stack (having extra method getMin() in O(1) time)

452- Sort Stack using an additional stack

453- Implement a queue which also provide get_max() along with push_back() and pop_front() all in O(1) time.

[Tough]

481- Largest rectangle under histogram

482- Finding max for sliding window of K size

Linked List:

[Easy]

500- Counting nodes in circular linked list ; Insert/Delete a node at the end/front of circular linked list

501- Merge 2 sorted linked list.

502- Reverse a linked list.

503- Find point of merger of two lists.

504- Remove duplicates in an unsorted list.

505- Return kth element from last ; using recursion

506- Delete middle element of the list

507- Check if list is palindrome

508- Add two integer numbers represented by linked list

509- Add two polynomials represented by linked list(each element having two variables 1- coefficient, 2- power )

510- Check if 2 lists are merging. Find point of merger

511- Split a Circular Linked List into two equal parts.

512- <JosephusCircle>: N people have decided to elect a leader by arranging themselves in a circle and eliminating every Mth person around the circle, closing ranks as each person drops out. Find which person will be the last one remaining (starting with rank 1).

513- Given a list, rotate the list to the right by k places, where k is non-negative. For example: Given 1_->2->3->4->5->NULL and k = 2, return 4->5->1->2->3->NULL.

514- A linked list has both even and odd numbers. Write an algorithm for making changes to the list in such a way that all even numbers appear at the beginning.

515- Flattening a doubly linked list with extra pointer of child list of same type. ; Unflatten new list to input list.

516- Remove middle element of linked list

517- Add to integer numbers represented as linked list.

518- Check if linked list is a palindrome

[Moderate]

551- <Interleaving starting half and reverse second half> Given a singly linked list L0 -> L1 -> … -> Ln-1 -> Ln.

Rearrange the nodes in the list so that the new formed list is : L0 -> Ln -> L1 -> Ln-1 -> L2 -> Ln-2 … Ex (1-> 2-> 3-> 4-> 5-> ===> 1-> 5-> 2-> 4-> 3)

552- Reverse list pairwise(1-> 2-> 3-> 4-> 5-> ===> 2-> 1-> 4-> 3-> 5 )

553- Sort a linked list

554- Swap linked list ith and jth elements from start and end by changing links

555- Partition a linked list by a number X.(X may have multiple entries in the list)

556- Detect position of loop in a linked list.Size of the loop.

557- Implement LRU cache

558- Merge K sorted list in a single sorted list[Heap]

559- Loop in a linked list; Size of the loop; Node where loop start

660- Reverse linked list in K size blocks

661- Input array of N integers is given. Compute all maximum value of sliding window of K size.

[Tough]

581- <DLL by XOR> Implement memory efficient double linked list using single linked list structure

582- <Clone a list with random pointer>

583-

Recursion:

[Easy]

601- merge 2 sorted linked list

602- reverse linked list pairwise

603- Print all string with valid parentheses n left and n right

603A- Print all string with n 0’s and n 1’s

603B- Print all n digit numbers with 1 surplus prefix

604- Find X^Y( X power Y)

605- Merge Sort

606- Quick Sort

607- Ways to score X by given shots(a, b, c, d etc)

607A- Numbers of ways to reach at Nth ladder’s step by taking 1, 2 or 3 steps only

608B- Numbers of ways to have money changes by given notes type only.

608 - Create minimum height BST from an sorted array

609- number of ways to reach to arr[n-1][m-1] from arr[0][0] by moving either right or down

610- paint fill method to fill a area by new color

611- Search word in a crossword of character matrix

[Moderate]

721- Regex match with . and * patterns

722- Count all inversion pairs in an integer array.

723- All Palindromic decompositions.

724- All permutations(input with or without repetitions)

725- All sub sets

726- All subsets of K elements from N size array

727- All positions of 8 not attacking queens in a 8*8 board

728- Return all possible IP addresses when dots( . ) got missing from original address( like 123.6.28.45  become  12362845 )

729- All valid permutations of n left and right parentheses ( )

730- All path of robot in a grid

731- Given a dictionary, a method to do lookup in dictionary and a M x N board where every cell has one character. Find all possible words that can be formed by a sequence of adjacent characters. Note that we can move to any of 4 adjacent characters, but a word should not have multiple instances of same cell.

732- Given an array of positive integers arr[] and a sum x, find all unique combinations in arr[] where the sum is equal to x.

[Tough]

771- Fill a Soduku game of 9*9 matrix

772- Sorted Linked List to height balanced BST in O(n) time

Heap:

[Easy]

801- There are given n ropes of different lengths, we need to connect these ropes into one rope. The cost to connect two ropes is equal to sum of their lengths. We need to connect the ropes with minimum cost

802- Merge n sorted arrays

803- Implement Stack/Queue using Heap

[Moderate]

821- Kth Maximun from a Max heap

822- K nearest element of median of given array

823- Find Kth smallest number from a row wise and column wise sorted 2D array.

[Tough]

840- Median in a stream of integers numbers

841- Compare Kth largest element of a max heap with a given numner

Sort/Search:

[Easy]

851- Min of rotated sorted array ; search in rotated sorted array

852- Index of turning number in UniModal array (unimodal array that increases than decreases like 1, 2, 3, 5, 7, 14, 11, 9, 8)

853- Find minimum sun array such that if you sort it entire array got sorted

[Moderate]

881- Find majority element in an array

882- Smallest k number of N size integer array

883- Magic index of a sorted array. Magic Index is one where Arr[i] == i

884- Minimum platforms needed to serve all trains schedule (minimum rooms for all meetings)

885- Minimum Visits to check all scheduled tasks in a factory

886- Maximum meetings you can attend in a day. All meetings start and end time is given

887- Merge Calendar meetings as disjoint busy and free time periods.

888- Sort a very big file like 20gb cant be loaded whole into RAM.

[Tough]

901 - Arrange given numbers to form the biggest number.

Hash:

[Easy]

951- remove repetitions in a linked list

[Moderate]

971- Implement LRU cache

972- 1st non-repeated number among a stream on integers

973- N points are given. Find line with highest points on it.

[Tough]

BinaryTree/BST:

[Easy]

1000- Return count/sum of all nodes/leafs/single parent/full nodes of a BT

1001- Height of Binary Tree

1002- <check clone BT> Check if two BT are some in terms of both content and structure

1003- Numbers of elements between (Low, High) values in a BST

1004- Return in-order successor of a node in BT, if parent pointer is given in all nodes

1005- Return post-order successor of a node in BT, if parent pointer is given in all nodes

1006- <Level Order Traversal> Create linked lists for each depth from a Btree or Print elements level wise, Level with max sum

1007- Print/Count all elements at K distance from the root of BT

1008- Check if BT is balanced or not.

1009- Check if given BT is BST or not

1010- Node with highest depth

1011- Sum of the multiplication of elements at each level

1012- Check a BT for the mirror image of another BT

1013-<In Order Successor> parent node is given

1014- Check if a BT is symmetrical i.e. its mirror image is same

1015- Connect all the adjacent nodes at the same level in a binary tree using extra nextRight pointer.

1016- Level with maximum nodes ; sum of nodes at each level s

[Moderate]

1101- Find diameter of a BT

1101- <First Common Ancestor> In a BT (easy in s BST)

1102- Implement getRandomNode() on a BT

1103- Preorder/Inorder traversal of BT without recursion. using explicit stack

1104- Create BT from inorder and preorder having unique elements

1105- Find max subtree which is full.

1106- Find max subtree which is Complete Binary Tree.

1107- Find max subtree of BT which is a BST.

1108- Find max path sum between any two nodes in a BT

1109- Serialize and de-serialize a BT

1110- Return random node of a BT. Random function is given

1111- Exterior of a binary tree. nodes in order => root to left most leaf -> leafs from left to right -> right most leaf to root

1112- Create BT from preorder and inorder traversal. All unique elements. ; without recursion

1113- Construct Binary Tree from given Parent Array representation

1114- Left View

1115- Top View

1116- Bottom View

1117- Leafs from left to right; layer by layer

1118- BT to doubly linked list

1119- Doubly Linked List to BT

1120- Create balanced BST from sorted array

1121- All possible BT from N identical nodes

1122- Maximum path sum between any two nodes in BT

1123- Maximum size BST in a BT

1124- Maximum size full tree in a BT

1125- Maximum size complete BT in a BT

[Tough]

1251- Print/return lists of all sequences which will create the same BST

1252- Check is a BT T2 is subtree of T1.

1253-<Count paths with sum K> In a BT return numbers of downwards paths of sum as K. A path can start or end at any node

1254- Postorder traversal of BT without recursion

1255- Print BT by level in a Zigzag order

1256- Clone a BT with random pointer

1257- Left View - Layer By Layer

1258- Right View - Layer By Layer

1259- Bottom View - Layer By Layer

1260- Merge two BST

1261- Create balanced BST from sorted linked list in O(N) time

1262- Two nodes of a BST are swapped, correct the BST

1263- Count of all Up to Down paths with sum equal to K.

Dynamic Programming:

[Easy]

1301- Number of ways to fill 2*N strip with 2*1 size bricks.

1302- Total ways to reach from left top to right bottom of a matrix by moving only right or down

[Moderate]

1351- <Matrix Chain Multiplication> Minimum cost of N Matrix multiplications (Printing brackets in Matrix Chain Multiplication Problem)

1352- Edit Distance between two strings

1353- Minimum number of coins for change

1354- Longest increasing subsequence

(1354B- Longest increasing subsequence’s length in NLogN time )

1355- <Russian doll problem> Stack of boxes

1356- Partition a set into two subsets such that the difference of subset sums is minimum

[Tough]

1341- Boolean evaluation: find no of ways to put parentheses to have true value for whole expression having AND, OR, and XOR operators

Greedy:

[Easy]

1501- Minimum platforms needed to serve all trains passing through a station. All trains arrival and departure time is given

1502- Build minimum road to connect all white buildings to black buildings in a line. There are n white and black buildings each. rode could be shared.

[Moderate]

[Tough]

Graph:

1601- DFS/BFS

1602- Shortest Distance between two nodes (dijkstra algorithm)

1603- Minimum spanning tree

1604- Topological sort

[Medium]

1651 - Given a sorted dictionary of an alien language, find order of characters.

Input: words[] = {"baa", "abcd", "abca", "cab", "cad"}

Output: Order of characters is 'b', 'd', 'a', 'c'

Bitwise/XOR:

[EASY]

1701- Find one non repeated number in an array.

1702- Count no of 1bits in the binary representation of a number

1703- Return max to two numbers without using comparison operator

1704- No of bits need to be flipped to change one integer into another

1705- Swap two numbers without any temp variable.

[MODERATE]

1751- Find two non repeated numbers in an array.

Miscellaneous Programming:

[Easy]

1801 - Find intersection point if exist for two lines given as point1 and point2.

1802- Find if 4 points are making square

1803- Find if two rectangles overlap. we are given following four coordinates.
 l1: Top Left coordinate of first rectangle. r1: Bottom Right coordinate of first rectangle.
 l2: Top Left coordinate of second rectangle. r2: Bottom Right coordinate of second rectangle.

Puzzles/Problems/Math/Probability/P&C/PS:

[Easy]

1- Number of trailing zeros in factorial N

[Moderate]

101- 100 doors. Flip every ith doors. Loop runs from 1 to 100. Initially all are closed. Tell which doors will be open in the end.

102- Bridge crossing — only two can cross at a time, minimize total time — 1, 2, 5, 10

103- 8 bolls and balance. find heavy boll

104- Fox and duck problem

105- Count 45 minutes when you have threads which take 1 hour to burn completely

106- 1000 bottles. one has poison

107- Cover chessboard with dominos. Two diagonal corners are already covered at start.

108- blue eyes people to leave island

109- 100 floor. 2 eggs. found min floor where egg will break

110- Find line which cut two squares in equal areas

111- 3 ants at the vertex of a triangle. they start moving randomly over edge with same speed. probability that no two ants collides.

112- Find 2nd best player among N players. You can have match between 2 players. Minimize total matches

113- 23 people are there in a room. Probability that at least two have birthday on same day. (No one has birthday on 29th fab)

114- 20 bottles. find bottle with heavy pills.

115- Boy to Girl ratio of a town where people keep having children till they have a girl child but no baby after a girl child.

116- 25 horses. only 5 can run in a race. minimum races you needed to find 3rd fastest horse

[Hard]

301- Josephus problem. N persons in a circle, every kth is eliminated repeatedly till only one left.

Puzzles -

Puzzles:

1- 25 horses . Find 3rd fastest

2- 100 doors .. flip in series 1 to 100

3- bridge crossing - 1, 2, 5, 10

4- find heavy boll , 25 bills

5- fox and duck

6- ant and tringle

7- 1000 bottles

8- black and white caps .. 100 prisoners

9- blue eye people

10- line to cut 2 squares into equal areas

12- boy and girl ratio in village

13- 23 people in room. Probability that at least two have Bday on same date. No 29th fan bday