Stacks and Queues:

- https://www.interviewbit.com/problems/simplify-directory-path/ *(contains many edge cases)*
- https://www.interviewbit.com/problems/largest-rectangle-in-histogram/ *(specific algorithm for stack approach, search google/youtube for better explanation)*
- https://www.interviewbit.com/problems/min-stack/ *(standard stack question, given in cormen)*
- https://www.interviewbit.com/problems/sliding-window-maximum/ *(tough to solve in O(n), read gfg article to understand deque approach)*
- https://www.geeksforgeeks.org/longest-monotonically-increasing-subsequence-size-n-log-n/ *(read $O(n^2)$ DP approach too)*
- Postfix, Prefix and Infix Conversions *(code not important as long as you know the algorithm)*

Two Pointers:

- Merge Two sorted arrays
- https://www.interviewbit.com/problems/minimize-the-absolute-difference/
- https://www.interviewbit.com/problems/3-sum/
- https://www.interviewbit.com/problems/diffk/
- https://www.interviewbit.com/problems/remove-duplicates-from-sorted-array-ii/
- https://www.interviewbit.com/problems/container-with-most-water/
- https://leetcode.com/problems/minimum-window-substring/ *(also read https://leetcode.com/problems/minimum-window-substring/discuss/26808/Here-is-a-10-line-template-that-can-solve-most-'substring'-problems)*
- https://leetcode.com/problems/find-all-anagrams-in-a-string/ *(also read https://leetcode.com/problems/find-all-anagrams-in-a-string/discuss/92007/Sliding-Window-algorithm-template-to-solve-all-the-Leetcode-substring-search-problem. Can be read later since map is used)*

Backtracking:

- https://www.interviewbit.com/problems/subset/ *(don't use python library functions in these questions)*

- https://www.interviewbit.com/problems/combination-sum-ii/
- https://www.interviewbit.com/problems/combination-sum/
- https://www.interviewbit.com/problems/palindrome-partitioning/ *(Read DP approach, frequently asked in MS interviews)*
- https://www.interviewbit.com/problems/generate-all-parentheses-ii/
- https://www.interviewbit.com/problems/nqueens/
- https://www.interviewbit.com/problems/sudoku/ *(read codes on leetcode discuss forum for nqueens and sudoku. Also, read problem and solution discussions on interviewbit if your solution is not getting accepted.)*
- https://www.interviewbit.com/problems/gray-code/ *(read the trick about how to generate next gray code)*
- https://www.geeksforgeeks.org/match-a-pattern-and-string-without-using-regular-expressions/ *(hashing required)*
- http://www.techiedelight.com/print-possible-knights-tours-chessboard/

Hashing:

- https://www.interviewbit.com/problems/largest-continuous-sequence-zero-sum/
- https://www.interviewbit.com/problems/4-sum/ *(try to also use two pointers)*
- https://www.interviewbit.com/problems/window-string/ *(similar to minimum window substring question that was covered in two-pointers)*
- https://www.interviewbit.com/problems/anagrams/ *(use sorting)*
- https://www.interviewbit.com/problems/points-on-the-straight-line/ *(store value as pair in map instead of double)*
- Read collision handling - open addressing and separate chaining
- https://www.geeksforgeeks.org/print-all-subarrays-with-0-sum/
- https://www.geeksforgeeks.org/count-distinct-elements-in-every-window-of-size-k

Trees:

- Preorder, Postorder, Inorder Traversal (both recursion and iterative)
- https://www.hackerrank.com/challenges/tree-height-of-a-binary-tree
- https://www.hackerrank.com/challenges/tree-top-view
- https://www.hackerrank.com/challenges/tree-level-order-traversal
- https://www.hackerrank.com/challenges/binary-search-tree-insertion
- https://www.hackerrank.com/challenges/tree-huffman-decoding

- https://www.hackerrank.com/challenges/binary-search-tree-lowest-common-ancestor
- https://www.hackerrank.com/challenges/is-binary-search-tree/problem
- https://www.hackerrank.com/challenges/swap-nodes-algo
- https://www.interviewbit.com/problems/identical-binary-trees/ *(check for edge cases)*
- https://www.interviewbit.com/problems/symmetric-binary-tree/
- https://www.interviewbit.com/problems/inorder-traversal-of-cartesian-tree/
- https://www.interviewbit.com/problems/sorted-array-to-balanced-bst/
- https://www.geeksforgeeks.org/construct-tree-from-given-inorder-and-preorder-traversal/
- https://www.geeksforgeeks.org/check-if-given-preorder-inorder-and-postorder-traversals-are-of-same-tree/
- https://www.geeksforgeeks.org/construct-tree-inorder-level-order-traversals-set-2/
- https://www.geeksforgeeks.org/full-and-complete-binary-tree-from-given-preorder-and-postorder-traversals/
- https://www.interviewbit.com/problems/kth-smallest-element-in-tree/
- https://www.interviewbit.com/problems/2sum-binary-tree/ *(hashing+tree)*
- https://www.interviewbit.com/problems/recover-binary-search-tree/ *(tough question, can skip this. read about morris traversal)*
- https://www.interviewbit.com/problems/invert-the-binary-tree/ *(easy recursion code)*
- https://www.interviewbit.com/problems/zigzag-level-order-traversal-bt/
- https://www.interviewbit.com/problems/populate-next-right-pointers-tree/
- https://www.interviewbit.com/problems/root-to-leaf-paths-with-sum/ *(backtracking)*

Graphs:

- https://www.interviewbit.com/problems/valid-path/ *(Simple BFS/DFS with bruteforce)*
- https://www.interviewbit.com/problems/commutable-islands/ *(Straightforward Kruskal Union-Find)*
- https://www.interviewbit.com/problems/possibility-of-finishing-all-courses-given-prerequisites/ *(Cycle Detection)*

- https://www.interviewbit.com/problems/black-shapes/ *(Number of connected components https://www.geeksforgeeks.org/find-number-of-islands/ https://www.geeksforgeeks.org/islands-in-a-graph-using-bfs/)*
- https://www.interviewbit.com/problems/capture-regions-on-board/ *(Flood Fill)*
- https://www.interviewbit.com/problems/largest-distance-between-nodes-of-a-tree/ *(read method from https://www.geeksforgeeks.org/longest-path-undirected-tree/)*
- https://www.interviewbit.com/problems/word-search-board/ *(read leetcode discussion solutions for hints and better codes https://leetcode.com/problems/word-search/discuss/ https://leetcode.com/problems/word-search-ii/description/)*
- https://www.interviewbit.com/problems/convert-sorted-list-to-binary-search-tree/ *(https://www.geeksforgeeks.org/sorted-linked-list-to-balanced-bst/)*
- https://www.interviewbit.com/problems/knight-on-chess-board/
- https://www.interviewbit.com/problems/word-ladder-i/ & https://www.interviewbit.com/problems/word-ladder-ii/
- Read
    - https://www.hackerearth.com/practice/algorithms/graphs/flood-fill-algorithm/tutorial/
    - https://www.hackerearth.com/practice/algorithms/graphs/shortest-path-algorithms/tutorial/
    - https://www.hackerearth.com/practice/algorithms/graphs/articulation-points-and-bridges/tutorial/
    - https://www.hackerearth.com/practice/algorithms/graphs/biconnected-components/tutorial/
    - https://www.hackerearth.com/practice/algorithms/graphs/strongly-connected-components/tutorial/
    - https://www.hackerearth.com/practice/algorithms/graphs/topological-sort/tutorial/
    - https://www.hackerearth.com/practice/data-structures/disjoint-data-strutures/basics-of-disjoint-data-structures/tutorial/

- https://www.geeksforgeeks.org/bipartite-graph/
- https://www.techiedelight.com/check-given-graph-strongly-connected-not/

- https://www.hackerrank.com/challenges/merging-communities
- https://www.hackerrank.com/challenges/components-in-graph

OOP

- Four pillars of OOP
- Why OOP?
- Encapsulation, Polymorphism, Inheritance, Abstraction
- Virtual Function (implementation in C++)
- Function and Operator Overloading (Examples)
- Abstract Class (Why can't we make objects of Abstract Class?)
- Exception handling
- Difference between Overloading and Overriding
- Dynamic / Run time polymorphism
- Static / Dynamic Binding
- Difference between Structure and Class
- Inheritance with different classifiers like public, protected, private
- https://hackernoon.com/the-top-10-object-oriented-design-interview-questions-developers-should-know-c7fc2e13ce39
- Virtual Memory and Paging
- Difference between void pointer and int pointer
- Immutable object
- Singleton Class
- Static, extern, final keywords
- Is java platform dependent? If not how is it achieved?
- Dangling pointers
- Different Design Questions

DBMS:

- SQL query involving joins which used group by and limit.
- Can Primary key contain two entities?
- Foreign Key
- ACID properties (Important)
- Entity integrity and Referential integrity
- Functional Dependency
- 1NF 2NF 3NF BCNF 4NF 5NF
- RDMS (One to One, One to many...)
- View and Trigger

- Transactions and indexing
- Deadlock conditions and how to solve it?
- Difference between SQL and No-SQL DBs
- Finding second largest number without using limits (nested queries)
- Questions to make database, normalizing the tables, solving queries on it..
- Indexing Structures, Primary, Secondary Indexes, Tree-structured Indexes, Hash-based Indexes
- Transaction management, serializability, recoverability, concurrency control, locking
- Criterion for Good Database Design
- ER to Relational Model
- Relational Calculus, Tuple Relational Calculus, Domain Relational Calculus