**A Step by Step Guide for Placement Preparation:-**

* Campus placements season is beginning at almost all the colleges and each one of us wants to prepare to do the best. A lot of students have been asking on different forums “How to use GeeksforGeeks for placement preparation”? In this article, a step by step guide for placement preparation is discussed. **Steps**
* **Aptitude Test**
  + [Course material](https://www.geeksforgeeks.org/placements-gq/)
  + [Practice Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
  + [Puzzles](https://www.geeksforgeeks.org/category/puzzles/)
* **Programming Language** It is recommended to know at least one language thoroughly)[**C**](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#C)**,**[**C++**](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#C++)**,**[**Java**](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#Java)**,**[**Python**](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#Python)
* **Subject basics :**Most important subjects that you must know for any company interview are undoubtedly data structures and algorithms.
  + [Data Structures](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#DS)
  + [Algorithms](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#Algorithms)
* **Other important CS subjects**: Please go through the basics of following three subjects :
  + [DBMS](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#DBMS)
  + [Operating System](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#Operating%20System)
  + [Computer Networks](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/#Computer%20Networks)
* [**Top topics for Interview Preparation for Software Developer**](https://www.geeksforgeeks.org/interview-preparation-for-software-developer/) : An assorted list of questions which should not be skipped.
* [**Tracking your progress**](https://www.geeksforgeeks.org/how-add-articles-to-to-do-and-done-list-on-geeksforgeeks/) : It is recommended that you create a profile on GeeksforGeeks for this. Logged in users can keep track of what all is done and what is in To Do.

1. [**Practice**](https://practice.geeksforgeeks.org/)**is undoubtedly the key to success** : Click [here](https://practice.geeksforgeeks.org/company-tags) and choose the company to practice the coding questions that have been asked earlier. ***Students generally are seen skipping***[***practice***](https://practice.geeksforgeeks.org/)***part and landing into trouble when asked to write the code in a live interview. Therefore, PRACTICE is strongly recommended.***
2. [**Top 25**](https://www.geeksforgeeks.org/top-25-interview-questions/)**–** This is the list of top 25 questions which have been asked in multiple companies.
3. [**MCQs of various subjects**](https://www.geeksforgeeks.org/quiz-corner-gq/): These are asked in first round of many companies and also help you build basics and foundations of any subject or topic.
4. [**Interview Corner**](https://www.geeksforgeeks.org/about/interview-corner/): Interview Experiences give you a fair idea about what you might go through in your interviews. You should carefully observe the comments wherein students have answered certain queries regarding placement process too. Choose your company and start preparing today!

[**Set 2**](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks-set-2/) – To guide you going beyond the basics and getting placed in your dream companies

**Topics:**

C

* [Course Material](https://www.geeksforgeeks.org/c/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Commonly asked Interview Questions](http://geeksquiz.com/commonly-asked-c-programming-interview-questions-set-1/)

C++

* [Course Material](https://www.geeksforgeeks.org/c-plus-plus/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Output based questions](https://www.geeksforgeeks.org/tag/output-of-cpp-program/)
* [Commonly asked Interview Questions](https://www.geeksforgeeks.org/commonly-asked-c-interview-questions-set-1/)

Java

* [Course Material](https://www.geeksforgeeks.org/java/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Output based questions](https://www.geeksforgeeks.org/tag/output-of-java-program/)
* [Commonly asked Interview Questions](https://www.geeksforgeeks.org/10-important-interview-questions/)

Python

* [Course Material](https://www.geeksforgeeks.org/python/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)

DS

* [Course Material](https://www.geeksforgeeks.org/data-structures/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Coding Practice](https://practice.geeksforgeeks.org/problems-level-page.php?level=0&offset=0&isPub=1)
* [Commonly asked Interview Questions](http://geeksquiz.com/commonly-asked-data-structure-interview-questions-set-1/)
* Most Important Topics
  + [Array](https://www.geeksforgeeks.org/data-structures/#Array), [Stack](https://www.geeksforgeeks.org/data-structures/#Stack) , [Queues](https://www.geeksforgeeks.org/category/queue/)
  + [Linked List](https://www.geeksforgeeks.org/data-structures/#LinkedList)
  + [Binary Search Trees](https://www.geeksforgeeks.org/data-structures/#BinarySearchTree)

Algorithms

* [Course Material](https://www.geeksforgeeks.org/fundamentals-of-algorithms/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/%20Mock%20Tests)
* [Coding practice](https://practice.geeksforgeeks.org/problems-level-page.php?level=1&offset=0&isPub=1)
* [Commonly asked Interview Questions](http://geeksquiz.com/commonly-asked-algorithm-interview-questions-set-1/)
* Most Important ones
  + [Searching and Sorting](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#SearchingandSorting)
  + [Divide and Conquer](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#DivideandConquer)
  + [Greedy Algorithms](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#GreedyAlgorithms)
  + [Dynamic Programming](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#DynamicProgramming)

DBMS

* [Course Material](https://www.geeksforgeeks.org/category/dbms/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Commonly asked Interview Questions](http://geeksquiz.com/commonly-asked-dbms-interview-questions/)

Computer Networks

* [Course Material](https://www.geeksforgeeks.org/category/computer-networks/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Commonly asked Interview Questions](https://www.geeksforgeeks.org/commonly-asked-computer-networks-interview-questions-set-1/)

Operating System

* [Course Material](https://www.geeksforgeeks.org/operating-systems/)
* [Practice MCQ Tests](https://www.geeksforgeeks.org/quiz-corner-gq/)
* [Commonly asked Interview Questions](https://www.geeksforgeeks.org/commonly-asked-operating-systems-interview-questions-set-1/)

If you still need more assistance with your placement preparation, have a look at our [Complete Interview Preparation Course](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?utm_source=practice&utm_medium=featured_list&utm_campaign=PRACTICE_ALL_COURSES_PAGE). The course has been designed by our expert mentors to help students **crack the coding interview of top product or service based organisations**. You get access to **premium lectures, 200+ coding questions bank, resume building tips, and lifetime access** to the course content. So to make sure that your next programming interview doesn’t feel like an interrogation, enroll in [Complete Interview Preparation](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?utm_source=practice&utm_medium=featured_list&utm_campaign=PRACTICE_ALL_COURSES_PAGE) and give a boost to your placement preparation. Now continue reading for part-2 of our Step by Step Guide that will help you in your placement preparation!

**Prerequisite**

Tech giants look for candidates who generally are excellent in algorithm designing and coding. You can surely **not** skip data structures and algorithms if you are looking to prepare for companies like Google, Facebook, Microsoft, Amazon, etc.

1. **Revise the working and use of popular Data Structures:** Apart from [set 1](https://www.geeksforgeeks.org/a-complete-step-by-step-guide-for-placement-preparation-by-geeksforgeeks/), more data structures to be focused are:
   * [Heap](https://www.geeksforgeeks.org/data-structures/#Heap)
   * [Hashing](https://www.geeksforgeeks.org/data-structures/#Hashing)
   * [Advanced Data Structures](https://www.geeksforgeeks.org/data-structures/#AdvancedDataStructure)
   * [Trees](https://www.geeksforgeeks.org/data-structures/#BinaryTree) and [Graphs](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#GraphAlgorithms)
2. **Time Complexities:**It is recommended to pay special attention to the analysis of extra space and time complexity of algorithms.
   * [Analysis of Algorithms](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#AnalysisofAlgorithms)
3. **Interesting Algorithms**: Mathematics is the base of algorithms and all those good at maths definitely play better with algorithms.
   * [Mathematical Algos](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#MathematicalAlgorithms)
   * [Randomized Algos](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#RandomizedAlgorithms)
   * [Bit Algos](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#BitAlgorithms)
   * [Pattern Searching](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#PatternSearching)
   * [Backtracking](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#Backtracking)
   * [String Algorithms](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#OtherStringAlgorithms)
   * [Branch and Bound](https://www.geeksforgeeks.org/fundamentals-of-algorithms/#BranchandBound)
4. [**System Design**](https://www.geeksforgeeks.org/category/design/)**Questions**: These questions are more likely to be asked to experienced candidates, but basic design questions like [OOP concepts](https://www.geeksforgeeks.org/commonly-asked-oop-interview-questions/), [Singleton Pattern](https://www.geeksforgeeks.org/singleton-design-pattern/), etc have been asked to freshers also.
5. [**MCQs**](https://www.geeksforgeeks.org/data-structures/#BinaryTree) : Although not directly asked in all companies, these multiple-choice questions help you make your concepts stronger and retain the concepts subconsciously.
6. [**Interview Corner**](https://www.geeksforgeeks.org/about/interview-corner/): Try to solve as many interview questions as you can for the company you are preparing. If finished, you can always practice other company questions too. If the company is coming for the internship, then prepare exclusively for [internship](https://www.geeksforgeeks.org/category/interview-experiences/internship-interview-experiences/)interviews.
7. [**GeeksforGeeks recommended list**](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/): This is the most important bunch of problems that you MUST solve yourself. Try each and every problem on the [practice platform](https://practice.geeksforgeeks.org/), discuss among each other for more efficient solutions. If you are not arriving at an efficient solution, **force yourself to solve it till the end irrespective of how lame your answer is** before looking at the complete solution at GeeksforGeeks. P.S: In onsite interviews **you will code on a whiteboard**, so make sure you practice doing that. It is very different from using an IDE.
8. **Projects you have done:**Thoroughlyrevise all the work you have done till now in your projects. The grilling about projects can sometimes be very deep. Also, choose your words before you speak. Mention only those topics where you think you are fine to be grilled upon
9. **Important FAQs** :
   * When to use which data structure & algorithm and why?
   * Why one approach is better than the other for a given problem?
   * The optimal solution for a given problem.
   * Which coding language is best for any given problem and why?
   * Real-life implementation of a given concept.
10. **Practice**[**topic wise**](https://practice.geeksforgeeks.org/topic-tags): You can pick topics one by one and practice them according to your weak areas.

If you still need more assistance with your placement preparation, have a look at our [Complete Interview Preparation Course](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?utm_source=CIP-A&utm_medium=Article&utm_campaign=a). The course has been designed by our expert mentors to help students **crack the coding interview of top product or** service-based organizations. You get access to **premium lectures, 200+ coding questions bank,**[**resume-building tips**](https://www.geeksforgeeks.org/resume-building-resources-and-tips/)**, and lifetime access** to the course content. So to make sure that your next programming interview doesn’t feel like an interrogation, enroll in [Complete Interview Preparation](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?utm_source=CIP-A&utm_medium=Article&utm_campaign=a) and give a boost to your placement preparation.

# Top 10 algorithms in Interview Questions

In this post “Top 10 coding problems of important  topics with their solutions ” are written. If you are preparing for a coding interview, going through these problems is a must.

**Topics :  
1.**[Graph](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo1) **2.**[Linked List](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo2) **3.**[Dynamic Programming](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo3) **4.**[Sorting And Searching](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo4) **5.**[Tree / Binary Search Tree](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo5) **6.**[Number Theory](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo6) **7.**[BIT Manipulation](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo7) **8.**[String / Array](https://www.geeksforgeeks.org/top-10-algorithms-in-interview-questions/?ref=lbp#algo8)

**Graph**

1. [Breadth First Search (BFS)](https://www.geeksforgeeks.org/breadth-first-traversal-for-a-graph/)  
2. [Depth First Search (DFS)](https://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/)  
3. [Shortest Path from source to all vertices \*\*Dijkstra\*\*](https://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/)  
4. [Shortest Path from every vertex to every other vertex \*\*Floyd Warshall\*\*](https://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/)  
5. [To detect cycle in a Graph \*\*Union Find\*\*](https://www.geeksforgeeks.org/union-find/)  
6. [Minimum Spanning tree \*\*Prim\*\*](https://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/)  
7. [Minimum Spanning tree \*\*Kruskal\*\*](https://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/)  
8. [Topological Sort](https://www.geeksforgeeks.org/topological-sorting/)  
9. [Boggle (Find all possible words in a board of characters)](https://www.geeksforgeeks.org/boggle-find-possible-words-board-characters/)  
10. [Bridges in a Graph](https://www.geeksforgeeks.org/bridge-in-a-graph/)

**Linked List**

1. [Insertion of a node in Linked List (On the basis of some constraints)](https://www.geeksforgeeks.org/given-a-linked-list-which-is-sorted-how-will-you-insert-in-sorted-way/)  
2. [Delete a given node in Linked List (under given constraints)](https://www.geeksforgeeks.org/delete-a-given-node-in-linked-list-under-given-constraints/)  
3. [Compare two strings represented as linked lists](https://www.geeksforgeeks.org/compare-two-strings-represented-as-linked-lists/)  
4. [Add Two Numbers Represented By Linked Lists](https://www.geeksforgeeks.org/sum-of-two-linked-lists/)  
5. [Merge A Linked List Into Another Linked List At Alternate Positions](https://www.geeksforgeeks.org/merge-a-linked-list-into-another-linked-list-at-alternate-positions/)  
6. [Reverse A List In Groups Of Given Size](https://www.geeksforgeeks.org/reverse-a-list-in-groups-of-given-size/)  
7. [Union And Intersection Of 2 Linked Lists](https://www.geeksforgeeks.org/union-and-intersection-of-two-linked-lists/)  
8. [Detect And Remove Loop In A Linked List](https://www.geeksforgeeks.org/detect-and-remove-loop-in-a-linked-list/)  
9. [Merge Sort For Linked Lists](https://www.geeksforgeeks.org/merge-sort-for-linked-list/)  
10. [Select A Random Node from A Singly Linked List](https://www.geeksforgeeks.org/select-a-random-node-from-a-singly-linked-list/)

**Dynamic Programming**

1. [Longest Common Subsequence](https://www.geeksforgeeks.org/dynamic-programming-set-4-longest-common-subsequence/)  
2. [Longest Increasing Subsequence](https://www.geeksforgeeks.org/dynamic-programming-set-3-longest-increasing-subsequence/)  
3. [Edit Distance](https://www.geeksforgeeks.org/dynamic-programming-set-5-edit-distance/)  
4. [Minimum Partition](https://www.geeksforgeeks.org/partition-a-set-into-two-subsets-such-that-the-difference-of-subset-sums-is-minimum/)  
5. [Ways to Cover a Distance](https://www.geeksforgeeks.org/count-number-of-ways-to-cover-a-distance/)  
6. [Longest Path In Matrix](https://www.geeksforgeeks.org/find-the-longest-path-in-a-matrix-with-given-constraints/)  
7. [Subset Sum Problem](https://www.geeksforgeeks.org/dynamic-programming-subset-sum-problem/)  
8. [Optimal Strategy for a Game](https://www.geeksforgeeks.org/dynamic-programming-set-31-optimal-strategy-for-a-game/)  
9. [0-1 Knapsack Problem](https://www.geeksforgeeks.org/dynamic-programming-set-10-0-1-knapsack-problem/)  
10. [Boolean Parenthesization Problem](https://www.geeksforgeeks.org/dynamic-programming-set-37-boolean-parenthesization-problem/)

**Sorting And Searching**

1. [Binary Search](http://geeksquiz.com/binary-search/)  
2. [Search an element in a sorted and rotated array](https://www.geeksforgeeks.org/search-an-element-in-a-sorted-and-pivoted-array/)  
3. [Bubble Sort](http://geeksquiz.com/bubble-sort/)  
4. [Insertion Sort](http://geeksquiz.com/insertion-sort/)  
5. [Merge Sort](http://geeksquiz.com/merge-sort/)  
6. [Heap Sort (Binary Heap)](http://geeksquiz.com/heap-sort/)  
7. [Quick Sort](http://geeksquiz.com/quick-sort/)  
8. [Interpolation Search](https://www.geeksforgeeks.org/interpolation-search/)  
9. [Find Kth Smallest/Largest Element In Unsorted Array](https://www.geeksforgeeks.org/kth-smallestlargest-element-unsorted-array-set-2-expected-linear-time/)  
10. [Given a sorted array and a number x, find the pair in array whose sum is closest to x](http://geeksquiz.com/given-sorted-array-number-x-find-pair-array-whose-sum-closest-x/)

**Tree / Binary Search Tree**

1. [Find Minimum Depth of a Binary Tree](https://www.geeksforgeeks.org/find-minimum-depth-of-a-binary-tree/)  
2. [Maximum Path Sum in a Binary Tree](https://www.geeksforgeeks.org/find-maximum-path-sum-in-a-binary-tree/)  
3. [Check if a given array can represent Preorder Traversal of Binary Search Tree](https://www.geeksforgeeks.org/check-if-a-given-array-can-represent-preorder-traversal-of-binary-search-tree/)  
4. [Check whether a binary tree is a full binary tree or not](https://www.geeksforgeeks.org/check-whether-binary-tree-full-binary-tree-not/)  
5. [Bottom View Binary Tree](https://www.geeksforgeeks.org/bottom-view-binary-tree/)  
6. [Print Nodes in Top View of Binary Tree](https://www.geeksforgeeks.org/print-nodes-top-view-binary-tree/)  
7. [Remove nodes on root to leaf paths of length < K](https://www.geeksforgeeks.org/remove-nodes-root-leaf-paths-length-k/)  
8. [Lowest Common Ancestor in a Binary Search Tree](https://www.geeksforgeeks.org/lowest-common-ancestor-in-a-binary-search-tree/)  
9. [Check if a binary tree is subtree of another binary tree](https://www.geeksforgeeks.org/check-binary-tree-subtree-another-binary-tree-set-2/)  
10. [Reverse alternate levels of a perfect binary tree](https://www.geeksforgeeks.org/reverse-alternate-levels-binary-tree/)

**Number Theory**

1. [Modular Exponentiation](https://www.geeksforgeeks.org/modular-exponentiation-power-in-modular-arithmetic/)  
2. [Modular multiplicative inverse](https://www.geeksforgeeks.org/multiplicative-inverse-under-modulo-m/)  
3. [Primality Test | Set 2 (Fermat Method)](https://www.geeksforgeeks.org/primality-test-set-2-fermet-method/)  
4. [Euler’s Totient Function](https://www.geeksforgeeks.org/eulers-totient-function/)  
5. [Sieve of Eratosthenes](https://www.geeksforgeeks.org/sieve-of-eratosthenes/)  
6. [Convex Hull](https://www.geeksforgeeks.org/convex-hull-set-1-jarviss-algorithm-or-wrapping/)  
7. [Basic and Extended Euclidean algorithms](https://www.geeksforgeeks.org/basic-and-extended-euclidean-algorithms/)  
8. [Segmented Sieve](https://www.geeksforgeeks.org/segmented-sieve/)  
9. [Chinese remainder theorem](https://www.geeksforgeeks.org/chinese-remainder-theorem-set-1-introduction/)  
10. [Lucas Theorem](https://www.geeksforgeeks.org/compute-ncr-p-set-2-lucas-theorem/)

**BIT Manipulation**

1. [Maximum Subarray XOR](https://www.geeksforgeeks.org/find-the-maximum-subarray-xor-in-a-given-array/)  
2. [Magic Number](https://www.geeksforgeeks.org/find-nth-magic-number/)  
3. [Sum of bit differences among all pairs](https://www.geeksforgeeks.org/sum-of-bit-differences-among-all-pairs/)  
4. [Swap All Odds And Even Bits](https://www.geeksforgeeks.org/swap-all-odd-and-even-bits/)  
5. [Find the element that appears once](https://www.geeksforgeeks.org/find-the-element-that-appears-once/)  
6. [Binary representation of a given number](https://www.geeksforgeeks.org/binary-representation-of-a-given-number/)  
7. [Count total set bits in all numbers from 1 to n](https://www.geeksforgeeks.org/count-total-set-bits-in-all-numbers-from-1-to-n/)  
8. [Rotate bits of a number](https://www.geeksforgeeks.org/rotate-bits-of-an-integer/)  
9. [Count number of bits to be flipped to convert A to B](https://www.geeksforgeeks.org/count-number-of-bits-to-be-flipped-to-convert-a-to-b/)  
10. [Find Next Sparse Number](https://www.geeksforgeeks.org/given-a-number-find-next-sparse-number/)

**String / Array**

1. [Reverse an array without affecting special characters](https://www.geeksforgeeks.org/reverse-an-array-without-affecting-special-characters/)  
2. [All Possible Palindromic Partitions](https://www.geeksforgeeks.org/given-a-string-print-all-possible-palindromic-partition/)  
3. [Count triplets with sum smaller than a given value](https://www.geeksforgeeks.org/count-triplets-with-sum-smaller-that-a-given-value/)  
4. [Convert array into Zig-Zag fashion](https://www.geeksforgeeks.org/convert-array-into-zig-zag-fashion/)  
5. [Generate all possible sorted arrays from alternate elements of two given sorted arrays](https://www.geeksforgeeks.org/generate-all-possible-sorted-arrays-from-alternate-elements-of-two-given-arrays/)  
6. [Pythagorean Triplet in an array](https://www.geeksforgeeks.org/find-pythagorean-triplet-in-an-unsorted-array/)  
7. [Length of the largest subarray with contiguous elements](https://www.geeksforgeeks.org/length-largest-subarray-contiguous-elements-set-1/)  
8. [Find the smallest positive integer value that cannot be represented as sum of any subset of a given array](https://www.geeksforgeeks.org/find-smallest-value-represented-sum-subset-given-array/)  
9. [Smallest subarray with sum greater than a given value](https://www.geeksforgeeks.org/minimum-length-subarray-sum-greater-given-value/)  
10. [Stock Buy Sell to Maximize Profit](https://www.geeksforgeeks.org/stock-buy-sell/)

1. [Amazon](https://www.geeksforgeeks.org/amazon-interview-preparation/)
2. [Samsung](https://www.geeksforgeeks.org/samsung-interview-preparation/)
3. [Microsoft](https://www.geeksforgeeks.org/microsoft-interview-preparation/)
4. [Adobe](https://www.geeksforgeeks.org/adobe-interview-preparation/)
5. [Oracle](https://www.geeksforgeeks.org/oracle-interview-preparation/)
6. [Google](https://www.geeksforgeeks.org/google-interview-preparation/)
7. [Flipkart](https://www.geeksforgeeks.org/flipkart-interview-preparation/)
8. [Facebook](https://www.geeksforgeeks.org/facebook-interview-preparation/)
9. [Goldman Sachs](https://www.geeksforgeeks.org/goldman-sachs-interview-preparation/)
10. [D E Shaw](https://www.geeksforgeeks.org/de-shaw-interview-preparation/)
11. [Cisco](https://www.geeksforgeeks.org/cisco-interview-preparation/)
12. [Visa](https://www.geeksforgeeks.org/visa-interview-preparation/)
13. [Paytm](https://www.geeksforgeeks.org/paytm-interview-preparation/)
14. [Morgan Stanley](https://www.geeksforgeeks.org/morgan-stanley-interview-preparation/)
15. [SAP Labs](https://www.geeksforgeeks.org/sap-labs-interview-preparation/)
16. [MAQ Software](https://www.geeksforgeeks.org/maq-software-interview-preparation/)
17. [Ola Cabs](https://www.geeksforgeeks.org/ola-cabs-interview-preparation/)
18. [VMware](https://www.geeksforgeeks.org/vmware-interview-preparation/)
19. [TCS](https://www.geeksforgeeks.org/tcs-recruitment-process/)
20. [Wipro](https://www.geeksforgeeks.org/wipro-recruitment-process/)
21. [Infosys](https://www.geeksforgeeks.org/infosys-recruitment-process/)
22. [IBM](https://www.geeksforgeeks.org/ibm-recruitment-process/)
23. [Hike](https://www.geeksforgeeks.org/hike-recruitment-process/)
24. [Cognizant](https://www.geeksforgeeks.org/cognizant-recruitment-process/)
25. [Nagarro](https://www.geeksforgeeks.org/nagarro-interview-process/)
26. [Directi](https://www.geeksforgeeks.org/directi-interview-preparation/)
27. [Ericson](https://www.geeksforgeeks.org/ericsson-interview-process/)
28. [Philips](https://www.geeksforgeeks.org/philips-interview-process/)
29. [Aricent](https://www.geeksforgeeks.org/aricent-recruitment-process/)
30. [NTT Data](https://www.geeksforgeeks.org/ntt-data-recruitment-process/)
31. [NEC](https://www.geeksforgeeks.org/nec-recruitment-process/)
32. [MakeMyTrip](https://www.geeksforgeeks.org/makemytrip-recruitment-process/)
33. [Zomato](https://www.geeksforgeeks.org/zomato-recruitment-process/)
34. [Opera](https://www.geeksforgeeks.org/opera-recruitment-process/)
35. [Expedia](https://www.geeksforgeeks.org/expedia-recruitment-process/)
36. [Intuit](https://www.geeksforgeeks.org/intuit-recruitment-process/)
37. [Snapdeal](https://www.geeksforgeeks.org/snapdeal-recruitment-process/)
38. [Qualcomm](https://www.geeksforgeeks.org/qualcomm-recruitment-process/)
39. [Linkedin](https://www.geeksforgeeks.org/linkedin-recruitment-process/)
40. [Kuliza](https://www.geeksforgeeks.org/kuliza-recruitment-process/)
41. [Yatra.com](https://www.geeksforgeeks.org/yatra-com-recruitment-process/)
42. [Citrix](https://www.geeksforgeeks.org/citrix-recruitment-process/)
43. [Arista](https://www.geeksforgeeks.org/arista-networks-recruitment-process/)
44. [Belzabar](https://www.geeksforgeeks.org/belzabar-recruitment-process/)
45. [CGI](https://www.geeksforgeeks.org/cgi-recruitment-process/)
46. [Brocade](https://www.geeksforgeeks.org/brocade-recruitment-process/)
47. [Oxygen Wallet](https://www.geeksforgeeks.org/oxygen-wallet-recruitment-process/)
48. [Amdocs](https://www.geeksforgeeks.org/amdocs-recruitment-process/)
49. [Zoho](https://www.geeksforgeeks.org/zoho-corporation-recruitment-process/)
50. [Nvidia](https://www.geeksforgeeks.org/nvidia-recruitment-process/)
51. [Accenture](https://www.geeksforgeeks.org/accenture-recruitment-process/)
52. [Accolite](https://www.geeksforgeeks.org/accolite-recruitment-process/)
53. [JUSPAY](https://www.geeksforgeeks.org/juspay-recruitment-process/)
54. [Walmart Labs](https://www.geeksforgeeks.org/walmart-labs-recruitment-process/)
55. [PayU](https://www.geeksforgeeks.org/payu-recruitment-process/)
56. [Thoughtworks](https://www.geeksforgeeks.org/thoughtworks-recruitment-process/)
57. [Intel](https://www.geeksforgeeks.org/intel-recruitment-process/)
58. [BankBazaar](https://www.geeksforgeeks.org/bankbazaar-recruitment-process/)
59. [BrowserStack](https://www.geeksforgeeks.org/browserstack-recruitment-process/)
60. [Groupon](https://www.geeksforgeeks.org/groupon-recruitment-process/)
61. [Grofers](https://www.geeksforgeeks.org/grofers-recruitment-process/)
62. [Tolexo](https://www.geeksforgeeks.org/tolexo-recruitment-process/)
63. [Synopsys](https://www.geeksforgeeks.org/synopsys-recruitment-process/)
64. [Yahoo](https://www.geeksforgeeks.org/yahoo-recruitment-process/)
65. [Deloitte](https://www.geeksforgeeks.org/deloitte-recruitment-process/)
66. [Symantec](https://www.geeksforgeeks.org/symantec-recruitment-process/)
67. [HCL](https://www.geeksforgeeks.org/hcl-recruitment-process/)
68. [Orange](https://www.geeksforgeeks.org/orange-recruitment-process/)
69. [UnitedHealth Group](https://www.geeksforgeeks.org/unitedhealth-group-recruitment-process/)
70. [Xome](https://www.geeksforgeeks.org/xome-recruitment-process/)
71. [Axtria](https://www.geeksforgeeks.org/axtria-recruitment-process/)
72. [Bidgely](https://www.geeksforgeeks.org/bidgely-recruitment-process/)
73. [Bloomberg](https://www.geeksforgeeks.org/bloomberg-recruitment-process/)
74. [HP](https://www.geeksforgeeks.org/hewlett-packard-recruitment-process/)
75. [Teradata](https://www.geeksforgeeks.org/teradata-recruitment-process/)
76. [Dell](https://www.geeksforgeeks.org/dell-recruitment-process/)
77. [Quikr](https://www.geeksforgeeks.org/quikr-recruitment-process/)
78. [OYO Rooms](https://www.geeksforgeeks.org/oyo-rooms-recruitment-process/)
79. [Practo](https://www.geeksforgeeks.org/practo-recruitment-process/)
80. [NHAI](https://www.geeksforgeeks.org/nhai-recruitment-process/)
81. [BARC](https://www.geeksforgeeks.org/barc-recruitment-process/)
82. [FactSet](https://www.geeksforgeeks.org/factset-recruitment-process/)
83. [Treebo](https://www.geeksforgeeks.org/treebo-recruitment-process/)
84. [Kronos](https://www.geeksforgeeks.org/kronos-incorporated-recruitment-process/)
85. [Swiggy](https://www.geeksforgeeks.org/swiggy-recruitment-process/)
86. [CouponDunia](https://www.geeksforgeeks.org/coupondunia-recruitment-process/)
87. [Arcesium](https://www.geeksforgeeks.org/arcesium-recruitment-process/)
88. [CarWale](https://www.geeksforgeeks.org/carwale-recruitment-process/)
89. [eBay](https://www.geeksforgeeks.org/ebay-recruitment-process/)
90. [Hewlett-Packard (HP)](https://www.geeksforgeeks.org/hewlett-packard-recruitment-process/)
91. [Atlassian](https://www.geeksforgeeks.org/atlassian-recruitment-process/)