**PLACEMENT PREPARATION**

* Please do practice and then see solution(don’t look at the solution immediately)
* Take 20 to 30 minutes(max) to solve problem. So think at least 15 minutes before seeing solution.

**ARRAYS**

* Maximum and Minimum element in an array using minimum comparisons.
  1. [Practice](http://practice.geeksforgeeks.org/problems/maximum-and-minimum-of-array-elements/0)
  2. [Solution](http://www.geeksforgeeks.org/maximum-and-minimum-in-an-array/)
* Kadane’s Algorithm
  1. [Practice](https://practice.geeksforgeeks.org/problems/kadanes-algorithm/0)
  2. [Solution](http://www.geeksforgeeks.org/largest-sum-contiguous-subarray/)
* Large Factorial (don’t use python :D)
  1. [Practice](http://practice.geeksforgeeks.org/problems/large-factorial/0)
  2. [Solution](http://www.geeksforgeeks.org/factorial-large-number/)
* GCD of array
  1. [Practice](http://practice.geeksforgeeks.org/problems/gcd-of-array/0)
  2. [Solution](http://practice.geeksforgeeks.org/editorial.php?pid=232)
* Consecutive one’s not allowed
  1. [Practice](http://practice.geeksforgeeks.org/problems/consecutive-1s-not-allowed/0)
  2. [Solution](http://www.geeksforgeeks.org/count-number-binary-strings-without-consecutive-1s/)
* Next smallest Palindrome
  1. [Practice](http://practice.geeksforgeeks.org/problems/next-smallest-palindrome/0)
  2. [Solution](http://www.geeksforgeeks.org/given-a-number-find-next-smallest-palindrome-larger-than-this-number/)
* Trapping Rain water
  1. [Practice](http://practice.geeksforgeeks.org/problems/trapping-rain-water/0)
  2. [Solution](http://www.geeksforgeeks.org/trapping-rain-water/)
* Distinct Even Numbers
  1. [Practice](http://practice.geeksforgeeks.org/problems/count-subsets-having-distinct-even-numbers/0)
  2. [Solution](http://www.geeksforgeeks.org/count-subsets-distinct-even-numbers/)
* [Maximum One's](http://practice.geeksforgeeks.org/problems/maximize-number-of-1s/0)
* [Zig-Zag](http://practice.geeksforgeeks.org/problems/convert-array-into-zig-zag-fashion/0)
* [Inversion array](http://practice.geeksforgeeks.org/problems/inversion-of-array/0)
* [Subarray sum](http://practice.geeksforgeeks.org/problems/subarray-with-given-sum/0)
* [Bitonic Subarray](http://practice.geeksforgeeks.org/problems/maximum-length-bitonic-subarray/0)
* [Tic Tac Toe](http://practice.geeksforgeeks.org/problems/tic-tac-toe/0)
* [Number to words](http://practice.geeksforgeeks.org/problems/number-to-words/0)
* [Possible Groups](http://practice.geeksforgeeks.org/problems/possible-groups/0)
* [Max and Min Product](http://practice.geeksforgeeks.org/problems/max-and-min-products/0)
* [Pairs with difference](http://www.geeksforgeeks.org/count-pairs-difference-equal-k/)
* [Chocolate distribution](http://practice.geeksforgeeks.org/problems/chocolate-distribution-problem/0)
* [Largest Subarray](http://practice.geeksforgeeks.org/problems/largest-subarray-with-gcd-one/0)
* [Four Number sum](http://practice.geeksforgeeks.org/problems/find-all-four-sum-numbers/0)
* [Largest Number](http://practice.geeksforgeeks.org/problems/form-largest-number-from-digits/0)
* [Equal Point](http://practice.geeksforgeeks.org/problems/find-equal-point-in-string-of-brackets/0)
* [Fill array](http://practice.geeksforgeeks.org/problems/fill-array-by-1s/0)
* [Stickler Thief](http://practice.geeksforgeeks.org/problems/stickler-theif/0)
* [Nth item sum](http://practice.geeksforgeeks.org/problems/nth-item-through-sum/0)
* [Common elements in arrays](http://practice.geeksforgeeks.org/problems/common-elements/0)
* [Largest array sum](http://practice.geeksforgeeks.org/problems/largest-sum-subarray-of-size-at-least-k/0)
* [sum equals to sum](http://practice.geeksforgeeks.org/problems/sum-equals-to-sum/0)
* [LIS in circular motion](http://practice.geeksforgeeks.org/problems/lis-in-circular-manner/0)

**STRINGS**

* [Anagram](http://practice.geeksforgeeks.org/problems/anagram/0)
* [KMP Implementation](http://practice.geeksforgeeks.org/problems/pattern-searching/0)
* [Braces](http://practice.geeksforgeeks.org/problems/parenthesis-checker/0)
* [First Non-Repeating Character](http://practice.geeksforgeeks.org/problems/non-repeating-character/0)
* [Longest sum string](http://practice.geeksforgeeks.org/problems/longest-even-length-substring/0)
* [Remove Duplicates](http://practice.geeksforgeeks.org/problems/remove-all-duplicates-from-a-given-string/0)
* [Longest Palindrome](http://practice.geeksforgeeks.org/problems/longest-palindrome-in-a-string/0)
* [All Permutations](http://practice.geeksforgeeks.org/problems/permutations-of-a-given-string/0) (see general implementation for this problem => no next\_permutation)
* [Remove Duplicates](http://practice.geeksforgeeks.org/problems/remove-duplicates/0) (O(1) space)
* [Betting](http://practice.geeksforgeeks.org/problems/betting-game/0)
* [Shortest direction](http://practice.geeksforgeeks.org/problems/shortest-direction/0)
* [Reverse Words](http://practice.geeksforgeeks.org/problems/reverse-words-in-a-given-string/0)
* [K-Pangram](http://practice.geeksforgeeks.org/problems/k-pangrams/0)
* [Longest Common Substring](http://practice.geeksforgeeks.org/problems/longest-common-substring/0) (not lcs)
* [Regex](http://practice.geeksforgeeks.org/problems/-regex-matching/0) , [Implement strstr](http://practice.geeksforgeeks.org/problems/implement-strstr/1) (KMP applications)
* [Roman to Integer](http://practice.geeksforgeeks.org/problems/roman-number-to-integer/0)
* [Check for Subsequence](http://practice.geeksforgeeks.org/problems/check-for-subsequence/0)
* [Substring subsequence](http://practice.geeksforgeeks.org/problems/substring-subsequence-problem/0)
* [Smallest Window](http://practice.geeksforgeeks.org/problems/smallest-window-in-a-string-containing-all-the-characters-of-another-string/0)
* [String Comparison](http://practice.geeksforgeeks.org/problems/string-comparison/0)
* [Longest Palindromic Subsequence](http://practice.geeksforgeeks.org/problems/longest-palindromic-subsequence/0)
* [Binary Multiplication](http://practice.geeksforgeeks.org/problems/karatsuba-algorithm/0)
* [Anagram of string](http://practice.geeksforgeeks.org/problems/anagram-of-string/1)
* [Run Length Encoding](http://practice.geeksforgeeks.org/problems/run-length-encoding/1)
* [Interleaved strings](http://practice.geeksforgeeks.org/problems/interleaved-strings/1)
* [Match Specific Pattern](http://practice.geeksforgeeks.org/problems/match-specific-pattern/1)
* [Grey Code sequence](http://practice.geeksforgeeks.org/problems/generate-grey-code-sequences/1)
* [Alien Dictionary](http://practice.geeksforgeeks.org/problems/alien-dictionary/1)
* [Ternary Expression](http://practice.geeksforgeeks.org/problems/convert-ternary-expression-to-binary-tree/1)
* [Modify string](http://practice.geeksforgeeks.org/problems/string-modification/1)
* [Generate IP addresses](http://practice.geeksforgeeks.org/problems/generate-ip-addresses/1)
* [Multiply two strings](http://practice.geeksforgeeks.org/problems/multiply-two-strings/1) (same as large factorial)
* [Palindromic Substring](http://practice.geeksforgeeks.org/problems/palindromic-patitioning/0)
* [Longest word in dictionary](http://practice.geeksforgeeks.org/problems/find-largest-word-in-dictionary/0)  (see trie concept also)

**System Design Questions (start preparing for this type of questions)**

[**https://www.educative.io/collection/5668639101419520/5649050225344512**](https://www.educative.io/collection/5668639101419520/5649050225344512)

[**https://hackernoon.com/anatomy-of-a-system-design-interview-4cb57d75a53f**](https://hackernoon.com/anatomy-of-a-system-design-interview-4cb57d75a53f)

**Linked List**

* [Segregate](http://practice.geeksforgeeks.org/problems/segregate-even-and-odd-nodes-in-a-linked-list/0)
* [reverse alternate](http://practice.geeksforgeeks.org/problems/given-a-linked-list-reverse-alternate-nodes-and-append-at-the-end/1)
* [Reverse in groups](http://practice.geeksforgeeks.org/problems/reverse-a-linked-list-in-groups-of-given-size/1)
* [Add two numbers](http://practice.geeksforgeeks.org/problems/add-two-numbers-represented-by-linked-lists/1)
* [Merge Two linked list](http://practice.geeksforgeeks.org/problems/merge-2-sorted-linked-list-in-reverse-order/1)
* [Detect and remove loop](http://practice.geeksforgeeks.org/problems/remove-loop-in-linked-list/1) (proof may be asked in F2F [proof](https://stackoverflow.com/a/16053582/5349104))
* [Middle element](http://practice.geeksforgeeks.org/problems/finding-middle-element-in-a-linked-list/1)
* [Delete without head](http://practice.geeksforgeeks.org/problems/delete-without-head-pointer/1)
* [Rearrange](http://practice.geeksforgeeks.org/problems/rearrange-linked-list-in-place/1http://practice.geeksforgeeks.org/problems/rearrange-linked-list-in-place/1)
* [Intersection](http://practice.geeksforgeeks.org/problems/intersection-point-in-y-shapped-linked-lists/1)
* [Flattening Linked list](http://practice.geeksforgeeks.org/problems/flattening-a-linked-list/1)
* [Remove every kth node](http://practice.geeksforgeeks.org/problems/remove-every-kth-node/1)
* [Implement stack](http://practice.geeksforgeeks.org/problems/implement-stack-using-linked-list/1) (see also queue)
* [Palindrome](http://practice.geeksforgeeks.org/problems/check-if-linked-list-is-pallindrome/1)
* [Union](http://practice.geeksforgeeks.org/problems/union-of-two-linked-list/1)
* [Alternate sort](http://practice.geeksforgeeks.org/problems/linked-list-that-is-sorted-alternatingly/1)
* [Clone with random pointer](http://practice.geeksforgeeks.org/problems/clone-a-linked-list-with-next-and-random-pointer/1) (see O(n) method)
* [Zig-Zag](http://practice.geeksforgeeks.org/problems/linked-list-in-zig-zag-fashion/1)
* [XOR](http://practice.geeksforgeeks.org/problems/xor-linked-list/1) (See concept alone)