Top Interview Questions

Want some extra practice? Here are a list of some of the top interview questions focusing on data structures and algorithms:  
  
#344 [Reverse String](https://leetcode.com/problems/reverse-string)

#412 [Fizz Buzz](https://leetcode.com/problems/fizz-buzz)

#136 [Single Number](https://leetcode.com/problems/single-number)

#104 [Maximum Depth of Binary Tree](https://leetcode.com/problems/maximum-depth-of-binary-tree)

#283 [Move Zeroes](https://leetcode.com/problems/move-zeroes)

#371 [Sum of Two Integers](https://leetcode.com/problems/sum-of-two-integers)

#206 [Reverse Linked List](https://leetcode.com/problems/reverse-linked-list)

#171 [Excel Sheet Column Number](https://leetcode.com/problems/excel-sheet-column-number)

#169 [Majority Element](https://leetcode.com/problems/majority-element)

#13 [Roman to Integer](https://leetcode.com/problems/roman-to-integer)

#237 [Delete Node in a Linked List](https://leetcode.com/problems/delete-node-in-a-linked-list)

#122 [Best Time to Buy and Sell Stock II](https://leetcode.com/problems/best-time-to-buy-and-sell-stock-ii)

#242 [Valid Anagram](https://leetcode.com/problems/valid-anagram)

#217 [Contains Duplicate](https://leetcode.com/problems/contains-duplicate)

#387 [First Unique Character in a String](https://leetcode.com/problems/first-unique-character-in-a-string)

#108 [Convert Sorted Array to Binary Search Tree](https://leetcode.com/problems/convert-sorted-array-to-binary-search-tree)

#268 [Missing Number](https://leetcode.com/problems/missing-number)

#350 [Intersection of Two Arrays II](https://leetcode.com/problems/intersection-of-two-arrays-ii)

#121 [Best Time to Buy and Sell Stock](https://leetcode.com/problems/best-time-to-buy-and-sell-stock)

#21 [Merge Two Sorted Lists](https://leetcode.com/problems/merge-two-sorted-lists)

#202 [Happy Number](https://leetcode.com/problems/happy-number)

#118 [Pascal's Triangle](https://leetcode.com/problems/pascals-triangle)

#70 [Climbing Stairs](https://leetcode.com/problems/climbing-stairs)

#101 [Symmetric Tree](https://leetcode.com/problems/symmetric-tree)

#53 [Maximum Subarray](https://leetcode.com/problems/maximum-subarray)

#326 [Power of Three](https://leetcode.com/problems/power-of-three)

#191 [Number of 1 Bits](https://leetcode.com/problems/number-of-1-bits)

#198 [House Robber](https://leetcode.com/problems/house-robber)

#66 [Plus One](https://leetcode.com/problems/plus-one)

#1 [Two Sum](https://leetcode.com/problems/two-sum)

#38 [Count and Say](https://leetcode.com/problems/count-and-say)

#26 [Remove Duplicates from Sorted Array](https://leetcode.com/problems/remove-duplicates-from-sorted-array)

#172 [Factorial Trailing Zeroes](https://leetcode.com/problems/factorial-trailing-zeroes)

#20 [Valid Parentheses](https://leetcode.com/problems/valid-parentheses)

#141 [Linked List Cycle](https://leetcode.com/problems/linked-list-cycle)

#234 [Palindrome Linked List](https://leetcode.com/problems/palindrome-linked-list)

#88 [Merge Sorted Array](https://leetcode.com/problems/merge-sorted-array)

#155 [Min Stack](https://leetcode.com/problems/min-stack)

#14 [Longest Common Prefix](https://leetcode.com/problems/longest-common-prefix)

#160 [Intersection of Two Linked Lists](https://leetcode.com/problems/intersection-of-two-linked-lists)

#28 [Implement strStr()](https://leetcode.com/problems/implement-strstr)

#69 [Sqrt(x)](https://leetcode.com/problems/sqrtx)

#190 [Reverse Bits](https://leetcode.com/problems/reverse-bits)

#125 [Valid Palindrome](https://leetcode.com/problems/valid-palindrome)

#189 [Rotate Array](https://leetcode.com/problems/rotate-array)

#204 [Count Primes](https://leetcode.com/problems/count-primes)

#7 [Reverse Integer](https://leetcode.com/problems/reverse-integer)

#94 [Binary Tree Inorder Traversal](https://leetcode.com/problems/binary-tree-inorder-traversal)

Amazon Interview Questions

If you would like to practice real life interview questions asked by Amazon, then here they are below. Keep in mind that it will be hard to just get everything right from the beginning. With enough practice you will become better and better, but there is an entire community of us learning, so I recommend you tackle these questions together with our online Discord community (see lesson #3 in this course for the link) and join the conversation and tackle problems in the #**interview-questions**channel:  
  
**1.** [Past Interview Questions](https://www.glassdoor.ca/Interview/Amazon-Software-Development-Engineer-Interview-Questions-EI_IE6036.0,6_KO7,36.htm)

**2.** *From Leetcode*:  
  
#1 [Two Sum](https://leetcode.com/problems/two-sum)

#2 [Add Two Numbers](https://leetcode.com/problems/add-two-numbers)

#3 [Longest Substring Without Repeating Characters](https://leetcode.com/problems/longest-substring-without-repeating-characters)

#200 [Number of Islands](https://leetcode.com/problems/number-of-islands)

#20 [Valid Parentheses](https://leetcode.com/problems/valid-parentheses)

#5 [Longest Palindromic Substring](https://leetcode.com/problems/longest-palindromic-substring)

#138 [Copy List with Random Pointer](https://leetcode.com/problems/copy-list-with-random-pointer)

#121 [Best Time to Buy and Sell Stock](https://leetcode.com/problems/best-time-to-buy-and-sell-stock)

#21 [Merge Two Sorted Lists](https://leetcode.com/problems/merge-two-sorted-lists)

**3.** **Bonus Question asked by Amazon:**

*From: https://www.dailycodingproblem.com/*

There's a staircase with N steps, and you can climb 1 or 2 steps at a time. Given N, write a function that returns the number of unique ways you can climb the staircase. The order of the steps matters.

For example, if N is 4, then there are 5 unique ways:

* 1, 1, 1, 1
* 2, 1, 1
* 1, 2, 1
* 1, 1, 2
* 2, 2

What if, instead of being able to climb 1 or 2 steps at a time, you could climb any number from a set of positive integers X? For example, if X = {1, 3, 5}, you could climb 1, 3, or 5 steps at a time. Generalize your function to take in X.

**Solution**

It's always good to start off with some test cases. Let's start with small cases and see if we can find some sort of pattern.

* N = 1: [1]
* N = 2: [1, 1], [2]
* N = 3: [1, 2], [1, 1, 1], [2, 1]
* N = 4: [1, 1, 2], [2, 2], [1, 2, 1], [1, 1, 1, 1], [2, 1, 1]

What's the relationship?

The only ways to get to N = 3, is to first get to N = 1, and then go up by 2 steps, or get to N = 2 and go up by 1 step. So f(3) = f(2) + f(1).

Does this hold for N = 4? Yes, it does. Since we can only get to the 4th step by getting to the 3rd step and going up by one, or by getting to the 2nd step and going up by two. So f(4) = f(3) + f(2).

To generalize, f(n) = f(n - 1) + f(n - 2). That's just the [Fibonacci sequence](https://en.wikipedia.org/wiki/Fibonacci_number)!

1. def staircase(n):
2. if n <= 1:
3. return 1
4. return staircase(n - 1) + staircase(n - 2)

Of course, this is really slow (O(2N)) — we are doing a lot of repeated computations! We can do it a lot faster by just computing iteratively:

1. def staircase(n):
2. a, b = 1, 2
3. for \_ in range(n - 1):
4. a, b = b, a + b
5. return a

Now, let's try to generalize what we've learned so that it works if you can take a number of steps from the set X. Similar reasoning tells us that if X = {1, 3, 5}, then our algorithm should be f(n) = f(n - 1) + f(n - 3) + f(n - 5). If n < 0, then we should return 0 since we can't start from a negative number of steps.

1. def staircase(n, X):
2. if n < 0:
3. return 0
4. elif n == 0:
5. return 1
6. else:
7. return sum(staircase(n - x, X) for x in X)

This is again, very slow (O(|X|N)) since we are repeating computations again. We can use dynamic programming to speed it up.

Each entry cache[i] will contain the number of ways we can get to step i with the set X. Then, we'll build up the array from zero using the same recurrence as before:

1. def staircase(n, X):
2. cache = [0 for \_ in range(n + 1)]
3. cache[0] = 1
4. for i in range(1, n + 1):
5. cache[i] += sum(cache[i - x] for x in X if i - x >= 0)
6. return cache[n]

This now takes O(N \* |X|) time and O(N) space.

Facebook Interview Questions

If you would like to practice real life interview questions asked by Facebook, then here they are below. Keep in mind that it will be hard to just get everything right from the beginning. With enough practice you will become better and better, but there is an entire community of us learning, so I recommend you tackle these questions together with our online Discord community (see lesson #3 in this course for the link) and join the conversation and tackle problems in the #**interview-questions**channel:  
  
**1.** [Past Facebook Interview Questions](https://www.glassdoor.ca/Interview/Facebook-Software-Engineer-Interview-Questions-EI_IE40772.0,8_KO9,26.htm)  
  
**2.** *From Leetcode:*

#1 [Two Sum](https://leetcode.com/problems/two-sum)

#200 [Number of Islands](https://leetcode.com/problems/number-of-islands)

#20 [Valid Parentheses](https://leetcode.com/problems/valid-parentheses)

#121 [Best Time to Buy and Sell Stock](https://leetcode.com/problems/best-time-to-buy-and-sell-stock)

#56 [Merge Intervals](https://leetcode.com/problems/merge-intervals)

#206 [Reverse Linked List](https://leetcode.com/problems/reverse-linked-list)

**3.** **Bonus Interview Question asked by Facebook:**

Determine the 10 most frequent words given a terabyte of strings.([solution](https://stackoverflow.com/questions/12525455/most-frequent-words-in-a-terabyte-of-data))

Google Interview Questions

If you would like to practice real life interview questions asked by Google, then here they are below. Keep in mind that it will be hard to just get everything right from the beginning. With enough practice you will become better and better, but there is an entire community of us learning, so I recommend you tackle these questions together with our online Discord community (see lesson #3 in this course for the link) and join the conversation and tackle problems in the #**interview-questions**channel:  
  
**1.** [Past Google Interview Questions](https://www.careercup.com/page?pid=google-interview-questions)  
  
**2.** *From Leetcode:*

#155 [Min Stack](https://leetcode.com/problems/min-stack)

#200 [Number of Islands](https://leetcode.com/problems/number-of-islands)

#20 [Valid Parentheses](https://leetcode.com/problems/valid-parentheses)

#42 [Trapping Rain Water](https://leetcode.com/problems/trapping-rain-water)

#56 [Merge Intervals](https://leetcode.com/problems/merge-intervals)

#681 [Next Closest Time](https://leetcode.com/problems/next-closest-time)

#139 [Word Break](https://leetcode.com/problems/word-break)

#31 [Next Permutation](https://leetcode.com/problems/next-permutation)

**3.** **Bonus Interview Question asked by Google:**

Write a function which will remove any given character from a String([solution](http://javarevisited.blogspot.sg/2015/04/how-to-remove-given-character-from.html) in Java)

Domain Specific Questions

With smaller companies, you may encounter more domain specific questions when it comes to coding interviews. What does this mean? It means that the questions focus less on computer science fundamentals like Data Structures and Algorithms, and instead they focus on the technology stack that the company is actively working with. To practice these questions, I have listed a great list for you to go through based on what type of job you are applying to (React developer, Mobile developer, etc...):  
  
<https://github.com/MaximAbramchuck/awesome-interview-questions>  
  
Also:

**Front End Developer:**<https://github.com/h5bp/Front-end-Developer-Interview-Questions>  
  
**Javascript Interview Questions:**[1](https://www.codementor.io/nihantanu/21-essential-javascript-tech-interview-practice-questions-answers-du107p62z), [2](https://www.toptal.com/javascript/interview-questions), [3](http://www.thatjsdude.com/interview/js2.html)  
  
PS, keep in mind that it will be hard to just get everything right from the beginning. With enough practice you will become better and better, but there is an entire community of us learning, so I recommend you tackle these questions together with our online Discord community (see lesson #3 in this course for the link) and join the conversation and tackle problems in the #**interview-questions**channel.