I'm human

Hello guys, welcome to this coding practice hub! If you're just starting out with programming or gearing up for a programming interview, you've come to the right place. I previously shared some awesome string, array, linked list, and binary tree problems, but now it's time to dive into common Java coding challenges tailored for beginners. These are the very same problems that helped me develop my coding skills and have appeared in many Java interviews. By tackling these puzzles, not only will you learn Java, but you'll also prepare yourself for those crucial Java interviews. Though simple at first glance, these programs delve deep into computer science principles, problem-solving strategies, data structures, algorithms, programming languages, control structures, and the art of transforming ideas into code – what I call 'coding sense.' This essential skill doesn't materialize overnight but develops gradually as you engage your brain with coding problems. Here are 50+ coding puzzles for Java programmers to practice, covering a range from simple checks (e.g., whether a number is even or odd) to more complex tasks (like checking if a given number is a palindrome). You can solve them in any order; I usually start with the easiest ones like calculating factorial. Once you've solved one problem, move on to the next. If you're stuck, try searching Google – sometimes, solutions are readily available. These problems will also test your basic knowledge of programming languages and control structures (e.g., loops, functions). You can learn these while trying to solve the puzzles, but remember to check the solution first if unsure. Some problems require familiarity with fundamental data structures such as arrays, linked lists, stacks, queues, binary search trees, graphs, etc. If you're not familiar or need a refresher on these concepts, consider checking out Udemy's Data Structures and Algorithms: Deep Dive Using Java course to get up to speed first. Don't worry about time; here are the coding problems for you to tackle – remember, practice makes perfect! 1. you need to tell whether they are a permutation of each other. 2. Given a string, print the first non-repeating char. 3. Given two sorted lists, combine them into one sorted list. 4. Insert a node in a doubly-linked list. 5. You are given an integer array of size n containing elements in the range 0 to n-1. Now count the frequency of all elements. 6. Given an array where every element occurs three times, except one element which occurs only once. Find the element that occurs once. 7. Given a positive integer n, count the total number of set bits in a binary representation of all numbers from 1 to n. 8. Write a function add() that returns the sum of two integers. The function should not use any of the arithmetic operators (+, ++, –, -, .. etc). 9. Write a Java program to find the largest or smallest of three integers, without using any of the comparison operators. 10. How to Print Fibonacci Series for N numbers 11. Write a Program to check if given number is a Armstrong Number 12. How to check if a number is Even or Odd 13. Java Program to find GCD of two numbers 14. How to Check if a number is a palindrome or not 15. How to Check if String is palindrome or not 16. Write a Java Program to calculate Simple interest 17. Write a Java Program to check a leap Year 18. How to implement Bubble Sort in Java 19. How to implement Insertion sort in Java 20. How to implement QuickSort in Java 21. Algorithm for Selection Sort 22. Algorithm for Merge Sort 23. Algorithm for Heap Sort 24. Java Program to count vowels and consonants in a String 25. How to reverse String in Java 26. Can you Code a Java Program to reverse a number 27. How to calculate Nth factorial in Java 28. How to check if a given number is a Prime number? 29. Write a program to calculate Area of a triangle in Java 1. Removing duplicates from an array in Java is a common problem where you can create a new array of unique elements using a set. 2. To find unique elements, use a set and copy array elements there; the set won't allow duplicates, giving you an array of unique elements. 3. For learning general coding techniques to solve problems like Sliding Window, fast and slow pointer, Dynamic Programming, check out Grokking the Coding Interview: Patterns for Coding Questions course on Educative. 4. 46. To determine if a number is a power of 4 or not, create a function that checks if the binary representation of the number has exactly one '1' bit. 5. 47. Compute the minimum or maximum of two integers without branching by using bitwise operators and comparing them directly. 6. 48. Given an unsigned integer, swap all odd bits with even bits in its binary representation. 7. 49. Find two numbers in an array where all but two are repeated once; use a hashmap to count occurrences of each number efficiently. 8. 50. Write an efficient C program to reverse the bits of a given number. 9. 51. Count the number of set bits in a number by using bitwise operators and iterating through the binary representation. 10. 52. Determine if two signed integers have different signs without using arithmetic operators; compare their most significant bits. 11. 53. Implement an LRU cache in Java using LinkedHashMap, which provides a notification method to evict the least recently used item when the cache becomes full.

Java programmeren voor beginners. Common java programs asked in selenium interview. Common string programs in java. Most common java programs asked in interview. Common java interview programs. Most common java programs. Common java programs asked in interview. Java programs for practice. Java programs. Popular java programs. Java programs based on command line arguments. Java programmeren. Common java code. 10 basic java programs.