Mock Interview Logical Implementation Questions

CSB DEVELOPMENTS

---

1. Find the Missing Number in an Array

Instructions:

You are given an array containing n - 1 numbers, where each number is in the range from 1 to n. The array contains all the numbers except one. Your task is to find the missing number without using extra space. Implement the solution efficiently.

Expected Output:

If the array is [3, 7, 1, 2, 8, 4, 5], the output should be 6.

---

2. Reverse a String Without Using Built-in Functions

Instructions:

You are given a string. Your task is to reverse the string without using any built-in functions such as reverse() or slicing. You need to reverse the string character by character.

Expected Output:

For input "software", the output should be "erawtfos".

---

3. Check If a Number Is Prime

Instructions:

You are given an integer. Your task is to check if the given number is a prime number or not. A prime number is a number greater than 1 that has no positive divisors other than 1 and itself.

Expected Output:

For input 17, the output should be True.

For input 4, the output should be False.

---

4. Find the First Non-Repeating Character in a String

Instructions:

You are given a string. Your task is to find the first character that does not repeat in the string. If all characters repeat, return None. The function should efficiently identify the first non-repeating character.

Expected Output:

For input "swiss", the output should be "w".

For input "aabcc" the output should be "b".

---

5. Fibonacci Sequence Using Recursion

Instructions:

You are given an integer n. Your task is to return the nth number in the Fibonacci sequence. The Fibonacci sequence is defined as follows:

- fib(0) = 0
- fib(1) = 1
- fib(n) = fib(n-1) + fib(n-2) for n > 1.

Expected Output:

For input 5, the output should be 5.

For input 10, the output should be 55.

---

6. Anagram Check

Instructions:

You are given two strings. Your task is to determine whether the two strings are anagrams of each other. Two strings are anagrams if they contain the same characters with the same frequencies.

Expected Output:

For input "listen" and "silent", the output should be True.

For input "hello" and "world", the output should be False.

---

7. Find the Longest Substring Without Repeating Characters

Instructions:

You are given a string. Your task is to find the length of the longest substring that does not contain any repeating characters. You need to return the length of this longest substring.

Expected Output:

For input "abcabcbb", the output should be 3 (The substring "abc").

For input "bbbbb", the output should be 1 (The substring "b").

---

8. Merge Two Sorted Arrays

Instructions:

You are given two sorted arrays. Your task is to merge them into a single sorted array. Write a function to combine both arrays, ensuring the result is sorted in ascending order.

Expected Output:

For input [1, 3, 5] and [2, 4, 6], the output should be [1, 2, 3, 4, 5, 6].

---

9. Count Vowels and Consonants in a String

Instructions:

You are given a string consisting of alphabets. Your task is to count the number of vowels and consonants in the string. Write a function that takes the string as input and returns the counts of vowels and consonants.

Expected Output:

For input "hello world", the output should be "Vowels: 3, Consonants: 7".

---

10. Implement a Queue Using Two Stacks

Instructions:

You are tasked with implementing a queue using two stacks. A queue follows the First In, First Out (FIFO) principle, while a stack follows Last In, First Out (LIFO). Your task is to simulate the behavior of a queue using two stacks. Implement both the enqueue and dequeue operations.

Expected Output:

For input enqueue(1), enqueue(2), and dequeue(), the output should be 1.

Then, for enqueue(3) and dequeue(), the output should be 2.

---

11. Sum of All Even Numbers in an Array

Instructions:

You are given an array of integers. Your task is to find and return the sum of all even numbers in the array. Write a function that loops through the array and sums up the even numbers.

Expected Output:

For input [1, 2, 3, 4, 5, 6], the output should be 12 (sum of 2, 4, and 6).

---

12. Check Palindrome

Instructions:

You are given a string. Your task is to check if the string is a palindrome. A palindrome is a word or phrase that reads the same backward as forward, ignoring spaces and punctuation.

Expected Output:

For input "racecar", the output should be True.

For input "hello", the output should be False.

---

13. Find the Intersection of Two Arrays

Instructions:

You are given two arrays. Your task is to find and return the intersection of these two arrays. The intersection of two arrays is a set of elements that appear in both arrays. Write a function that returns the intersection of the arrays.

Expected Output:

For input [1, 2, 2, 1] and [2, 2], the output should be [2].

For input [4, 9, 5] and [9, 4, 9, 8, 4], the output should be [9, 4].

---

14. Find the Missing Element in an Array of Consecutive Numbers

Instructions:

You are given an array that contains n - 1 integers, where the integers are from 1 to n. The array is sorted in ascending order, and there is exactly one number missing from the sequence. Your task is to find the missing number.

Expected Output:

For input [1, 2, 4, 5, 6], the output should be 3.

---

15. Reverse Words in a String

Instructions:

You are given a string with several words separated by spaces. Your task is to reverse the order of the words in the string. Do not reverse the individual characters within the words.

Expected Output:

For input "hello world", the output should be "world hello".

---

16. Calculate the Factorial of a Number

Instructions:

You are given an integer n. Your task is to compute the factorial of n, which is the product of all positive integers less than or equal to n. Write a function that calculates the factorial of a given number.

Expected Output:

For input 5, the output should be 120 (because 5! = 5 × 4 × 3 × 2 × 1 = 120).

---

17. Implement a Stack Using Two Queues

Instructions:

You are tasked with implementing a stack using two queues. A stack follows Last In, First Out (LIFO), while a queue follows First In, First Out (FIFO). Your task is to implement the push and pop operations of a stack using two queues.

Expected Output:

For input push(1), push(2), and pop(), the output should be 2.

Then, for push(3) and pop(), the output should be 3.

---

18. Check for Balanced Parentheses in an Expression

Instructions:

You are given a string that contains parentheses. Your task is to check if the parentheses are balanced. A string has balanced parentheses if each opening parenthesis has a corresponding closing parenthesis in the correct order.

Expected Output:

For input "()", the output should be True.

For input "(()", the output should be False.

---

19. Count the Number of Set Bits in an Integer

Instructions:

You are given an integer. Your task is to count the number of set bits (1s) in its binary representation. Write a function that computes the number of set bits in the integer.

Expected Output:

For input 5 (binary 101), the output should be 2 (because there are two set bits).

---

20. Find the Largest Number in an Array Without Sorting

Instructions:

You are given an array of integers. Your task is to find the largest number in the array without sorting it. Write a function that finds the maximum number by iterating through the array.

Expected Output:

For input [1, 3, 5, 2, 4], the output should be 5.

Logical Implementation Questions (Long)

---

1. Find the Missing Number in an Array

Instructions:

Imagine you are working on an application where you need to process a large dataset of consecutive numbers ranging from 1 to n. However, for some reason, one of the numbers is missing from the dataset. You are given an array of n - 1 integers, all of which are distinct, and the numbers in the array range from 1 to n. The challenge is to identify the missing number efficiently. The array is unsorted, and you cannot use extra space for storing data. Write a function to find the missing number in the range using a mathematical approach or an efficient algorithm that doesn't require sorting or additional storage.

Expected Output:

If the input array is [3, 7, 1, 2, 8, 4, 5], the function should return 6, as it is the missing number in the sequence ranging from 1 to 8.

---

2. Reverse a String Without Using Built-in Functions

Instructions:

In many applications, strings are frequently manipulated. One common operation is to reverse a string. However, you are working on an environment where you are not allowed to use built-in functions like reverse() or slicing features to reverse strings. Your task is to write a function that takes a string as input and returns the string reversed. You are to implement this reversal manually by iterating over the string and manipulating its characters to create the reversed version.

Expected Output:

For input "software", the expected output should be "erawtfos", which is the reversed version of the input string.

---

3. Check If a Number Is Prime

Instructions:

Prime numbers are fundamental to many areas of mathematics, especially in cryptography and number theory. In this problem, you are tasked with determining whether a given number is prime. A prime number is a positive integer greater than 1 that has no positive divisors other than 1 and itself. You need to write a function that efficiently checks if a given number is prime. This function should be optimized to avoid unnecessary computations, especially for larger numbers.

Expected Output:

For input 17, the function should return True, as 17 is a prime number. For input 4, the output should be False, since 4 is divisible by 2.

---

4. Find the First Non-Repeating Character in a String

Instructions:

In many string processing tasks, you need to identify specific characters within the string based on certain criteria. One such task is finding the first character in a string that doesn't repeat. Given a string, you are to write a function that returns the first character that does not repeat. If all characters repeat, the function should return None. The challenge is to implement this in an efficient way, so you don't unnecessarily traverse the string multiple times.

Expected Output:

For input "swiss", the first non-repeating character is "w". The function should return "w". For input "aabcc", the function should return "b", since it's the first non-repeating character.

---

5. Fibonacci Sequence Using Recursion

Instructions:

The Fibonacci sequence is a famous sequence of numbers in which each number is the sum of the two preceding ones, starting from 0 and 1. In this problem, you are tasked with implementing a function that takes an integer n and returns the nth Fibonacci number. The Fibonacci sequence is defined as follows:

- `fib(0) = 0`
- `fib(1) = 1`
- `fib(n) = fib(n-1) + fib(n-2) for n > 1`

  You are asked to solve this problem using recursion, which is a natural way to express the Fibonacci relation.

Expected Output:

For input 5, the output should be 5 since the Fibonacci sequence up to the 5th element is 0, 1, 1, 2, 3, 5. For input 10, the output should be 55, since the 10th Fibonacci number is 55.

---

6. Anagram Check

Instructions:

An anagram is a word or phrase that is formed by rearranging the letters of another word or phrase, using all the original letters exactly once. You are given two strings, and your task is to determine whether these two strings are anagrams of each other. The strings may contain spaces, so you should ignore them when checking for anagrams. Additionally, you need to ensure that the frequency of each character is the same in both strings, regardless of order. Write a function to check if the two strings are anagrams.

Expected Output:

For input "listen" and "silent", the function should return True, as both are anagrams of each other. For input "hello" and "world", the output should be False, since they are not anagrams.

---

7. Find the Longest Substring Without Repeating Characters

Instructions:

In many applications, it is useful to identify substrings that follow certain patterns. One common problem is finding the longest substring without repeating characters. Given a string, your task is to find the longest substring that contains no repeated characters. This problem can be solved efficiently using a sliding window technique. The challenge is to implement a function that does this in linear time, without checking every possible substring.

Expected Output:

For input "abcabcbb", the function should return 3 because the longest substring without repeating characters is "abc". For input "bbbbb", the function should return 1, as the longest substring without repeating characters is "b".

---

8. Merge Two Sorted Arrays

Instructions:

You are given two sorted arrays. Your task is to merge these arrays into a single sorted array. You cannot use built-in functions like sort(), and the solution should be implemented efficiently. Write a function that combines both arrays and ensures that the result is sorted in ascending order. This is a common operation in applications that process large datasets, and it requires a merge algorithm similar to the one used in the merge sort algorithm.

Expected Output:

For input [1, 3, 5] and [2, 4, 6], the output should be [1, 2, 3, 4, 5, 6], combining both sorted arrays into one sorted array.

---

9. Count Vowels and Consonants in a String

Instructions:

Given a string consisting of alphabetic characters, your task is to count the number of vowels and consonants in the string. Vowels are considered to be the characters a, e, i, o, u (case-insensitive), while all other alphabetic characters are consonants. You should write a function that counts the vowels and consonants in the string and returns the results.

Expected Output:

For input "hello world", the output should be "Vowels: 3, Consonants: 7", as the string contains 3 vowels (e, o, o) and 7 consonants (h, l, l, w, r, l, d).

---

10. Implement a Queue Using Two Stacks

Instructions:

A queue is a data structure that follows the **First In, First Out (FIFO)** principle, while a stack follows the **Last In, First Out (LIFO)** principle. In this problem, you need to implement a queue using two stacks. This problem can be solved by using two stacks where one stack is used for enqueue operations and the other for dequeue operations. Write a function that simulates the behavior of a queue using two stacks.

Expected Output:

For the sequence of operations: enqueue(1), enqueue(2), dequeue(), the output should be 1. Then, for enqueue(3) and dequeue(), the output should be 2.

---

11. Sum of All Even Numbers in an Array

Instructions:

You are given an array of integers. Your task is to find and return the sum of all even numbers in the array. This problem requires a simple iteration through the array and checking if each number is even. You should then sum up the even numbers and return the result.

Expected Output:

For input [1, 2, 3, 4, 5, 6], the sum of even numbers is 12 because the even numbers are 2, 4, 6.

---

12. Check Palindrome

Instructions:

A palindrome is a word, phrase, or sequence that reads the same backward as forward, ignoring spaces, punctuation, and capitalization. Your task is to write a function that checks if a given string is a palindrome. The function should handle strings with spaces, punctuation, and mixed case.

Expected Output:

For input "racecar", the output should be True. For input "hello", the output should be False.