Stack Problems for the Day Before Your Coding Interview

Stacks are a fundamental data structure in computer science, and they are often tested in coding interviews. If you are preparing for a coding interview, it is important to be able to solve stack problems efficiently and correctly.

In this article, we will cover some of the most common stack problems that you may encounter in a coding interview. We will also provide practice problems and solutions to help you prepare.

Stack Problems for the day before your Coding Interview (Day before Coding Interview Book 9)

by Aditya Chatterjee

4.8 out of 5

Language	: English
File size	: 1466 KB
Text-to-Speech	: Enabled
Screen Reader	: Supported
Enhanced typesetting	: Enabled
Print length	: 56 pages
Lending	: Enabled

Stack Basics

A stack is a linear data structure that follows the last-in, first-out (LIFO) principle. This means that the last element that is added to the stack will be the first element that is removed.

Stacks are commonly implemented using arrays or linked lists. However, for the purposes of this article, we will focus on stack implementations using arrays.

Array-Based Stack Implementation

An array-based stack can be implemented using an array of fixed size. The following code shows an example of an array-based stack implementation in JavaScript:

```
class Stack { constructor(size){this.array = new Array(size); this.top = -1; }
```

```
push(element){if (this.top === this.array.length - 1){throw new Error("Stack overflow"); }
```

```
this.top++; this.array[this.top] = element; }
```

```
pop(){if (this.top === -1){throw new Error("Stack underflow"); }
```

```
const element = this.array[this.top]; this.top--; return element; }
```

```
peek(){if (this.top === -1){return null; }
```

```
return this.array[this.top]; }
```

```
isEmpty(){return this.top === -1; }
```

```
isFull(){return this.top === this.array.length - 1; }}
```

The `push()` method adds an element to the top of the stack. The `pop()` method removes and returns the element at the top of the stack. The `peek()` method returns the element at the top of the stack without removing it. The `isEmpty()` method returns `true` if the stack is empty and `false` otherwise. The `isFull()` method returns `true` if the stack is full and `false` otherwise.

Common Stack Problems

Now that we have covered the basics of stacks, let's take a look at some of the most common stack problems that you may encounter in a coding interview.

1. Balanced Parentheses

The Balanced Parentheses problem is a classic stack problem. The problem asks you to determine whether a string of parentheses is balanced. A string of parentheses is balanced if every opening parenthesis has a corresponding closing parenthesis.

For example, the following string of parentheses is balanced:

```
()()()
```

However, the following string of parentheses is not balanced:

```
)()()
```

Here is a possible solution to the Balanced Parentheses problem in JavaScript:

```
function isBalanced(string){const stack = new Stack();
```

```
for (let i = 0; i
```

2. Infix to Postfix Conversion

The Infix to Postfix Conversion problem is another common stack problem. The problem asks you to convert an infix expression to a postfix expression. An infix expression is an expression that uses infix operators, such as `+`, `-`, `*`, and `/`. A postfix expression is an expression that uses postfix operators, such as `a+b`, `a-b`, `a*b`, and `a/b`.

For example, the following infix expression can be converted to the following postfix expression:

```
(a + b) * c
```

```
ab+c*
```

Here is a possible solution to the Infix to Postfix Conversion problem in JavaScript:

```
function infixToPostfix(infixExpression){const stack = new Stack();
const output = [];
```

```
for (let i = 0; i
```

3. Tower of Hanoi

The Tower of Hanoi problem is a classic stack problem. The problem asks you to move a stack of disks from one tower to another tower, using a third tower as an intermediate. The disks are of different sizes, and the larger disks cannot be placed on top of the smaller disks.

For example, the following is a representation of the Tower of Hanoi problem with three disks:

```
A: B: C: 1 2 3
```

The goal is to move the disks from tower C to tower A, using tower B as an intermediate.

Here is a possible solution to the Tower of Hanoi problem in JavaScript:

```
function towerOfHanoi(numDisks, fromTower, toTower, auxTower){if (numDisks === 1){console.log(`Move disk 1 from ${fromTower}to ${toTower}`); return; }
```

```
towerOfHanoi(numDisks - 1, fromTower, auxTower, toTower); console.log(`Move disk ${numDisks}from ${fromTower}to ${toTower}`); towerOfHanoi(numDisks - 1, auxTower, toTower, fromTower); }
```

Practice Problems

In addition to the problems covered in this article, there are many other stack problems that you may encounter in a coding interview. Here are a few practice problems to help you prepare:

1. `Reverse a stack.`
2. `Sort a stack using a temporary stack.`
3. `Evaluate a postfix expression.`
4. `Find the maximum element in a stack in O(1) time.`
5. `Implement a queue using two stacks.`

`Stacks are a fundamental data structure in computer science, and they are often tested in coding interviews. By understanding the basics of stacks and practicing the common stack problems covered in this article, you can improve your chances of success in your`

Stack Problems for the day before your Coding Interview (Day before Coding Interview Book 9)

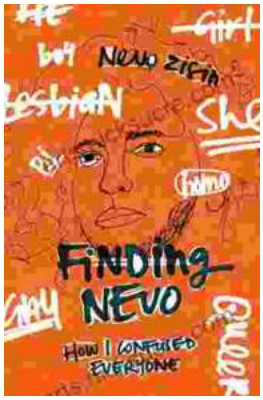
by Aditya Chatterjee

4.8 out of 5

Language	: English
File size	: 1466 KB
Text-to-Speech	: Enabled
Screen Reader	: Supported
Enhanced typesetting	: Enabled
Print length	: 56 pages
Lending	: Enabled

The Ultimate Canadian Cookbook: A Culinary Exploration of Iconic Dishes and Regional Flavors

Journey into the heart of Canadian cuisine with "The Ultimate Canadian Cookbook," an indispensable culinary guide that unveils the vibrant flavors, diverse traditions, and...



Finding Nevo: Unraveling the Mysterious Discography that Confused Everyone

A Fragmentary Puzzle In the labyrinthine world of music, there exists an enigmatic figure known only as Nevo. Their...