DSA Tutorial

Stacks

20-27 January 2025

Theory

Discuss the *LIFO* property of `Stacks` and different ADT methods along with their time complexity :

- `push`
- `pop`
- `top` / `peek`

Implementation

- Make sure that the students can understand `Stack` as a linked list of integers.
- How to reduce the time complexity of calculating stack size from $O(n)$ to $O(1)$? A possible solution : Create a *struct*, with members `stack` and `size`, have the original linked list's head stored in `stack` and update the `size` variable when performing `push` or `pop`.

Questions

Validate Stack Sequences

Given two integer arrays `pushed` and `popped` each with distinct values, return true if this could have been the result of a sequence of push and pop operations on an initially empty stack, or false otherwise. More information : Leetcode

Count Substrings

You are given a string s and a character c. Return the total number of substrings of s that start and end with c. More information : Leetcode

Ideally, spend 15 minutes discussing the theory. Then discuss both the problems and their pseudo code (10 minutes they can think, explain the idea and pseudo code in 10 minutes for each problem), otherwise, if possible, discuss just the ideas and the reasoning behind using stacks (We can work with just the three operations that stacks have).