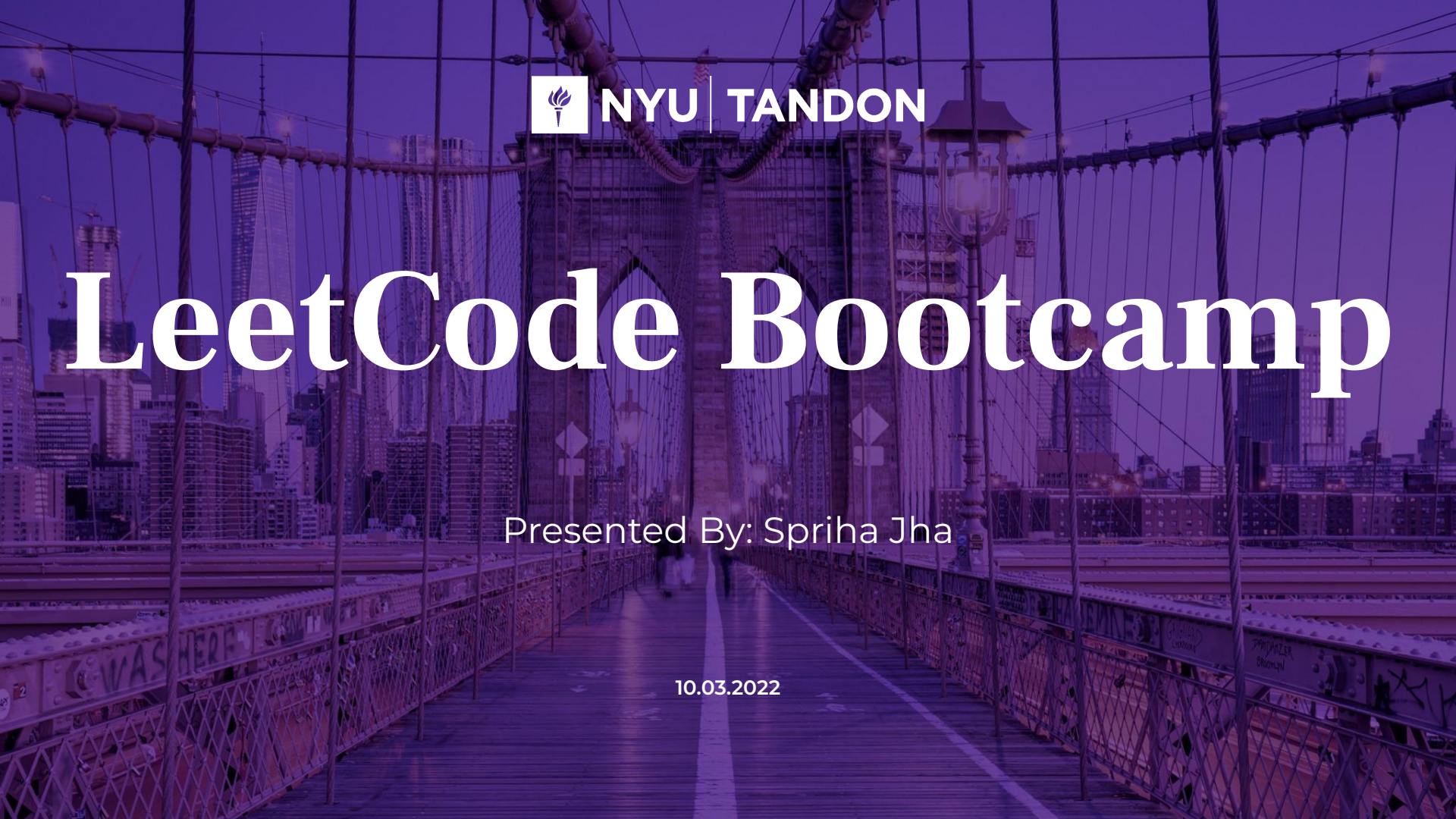


LeetCode Bootcamp

Presented By: Spriha Jha

10.03.2022

Session Outline

01. LeetCode platform overview

02. Tech Interview Resources

03. Bootcamp Timeline

04. Introduction to Arrays & Sorting

05. Problem Sets

06. Debrief & Q/A

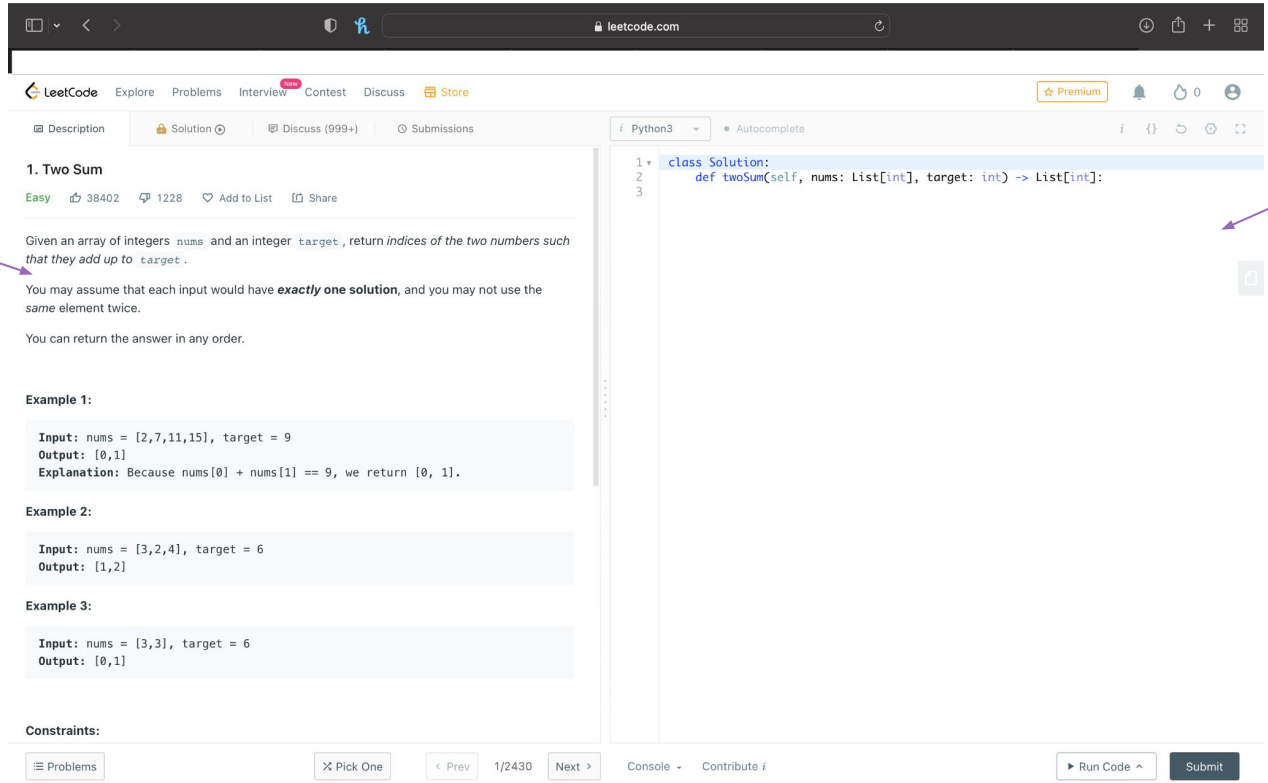
NYU TANDON

PART 01

LeetCode Platform Overview

Free Tier

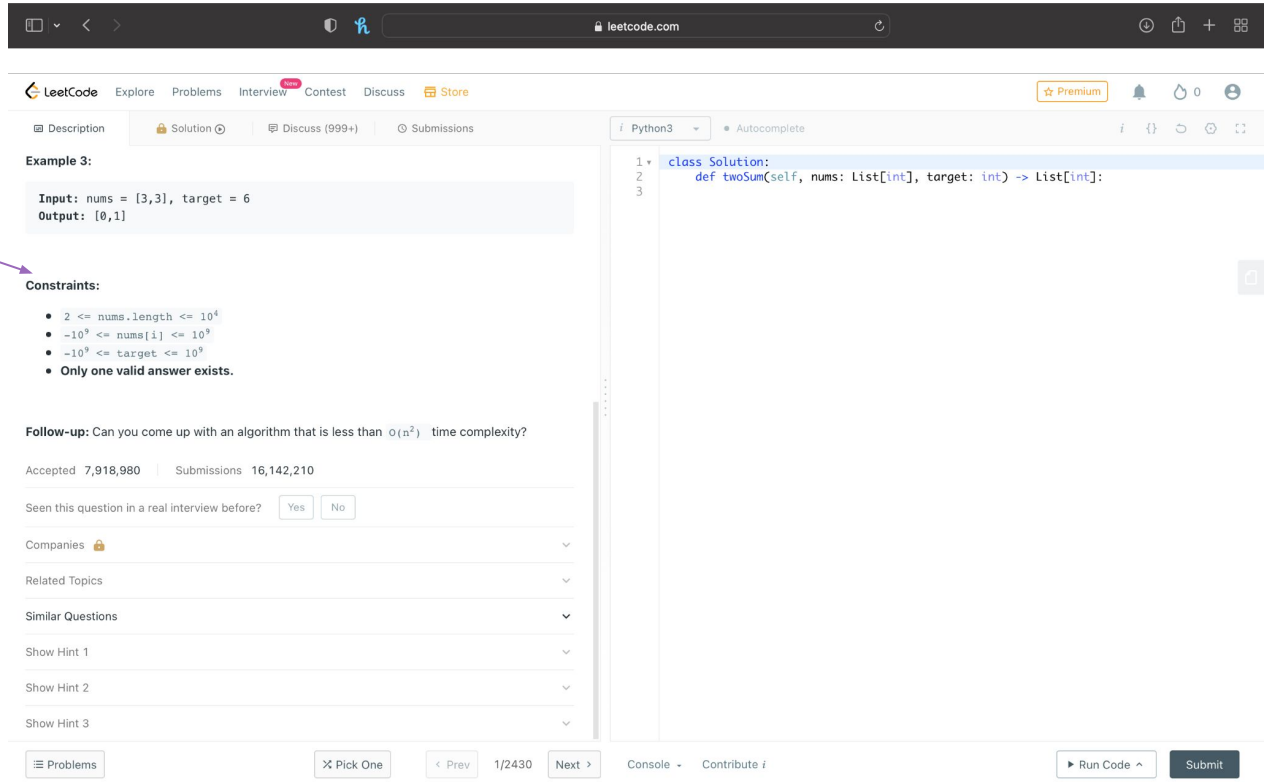
Problem Statement



Add your code here!

NYU TANDON

Identifying edge-cases



NYU TANDON

PART 02

Tech Interview Resources

Grind75 | CodeSignal | GitHub

Tech Interview Handbook

techinterviewhandbook.org

GRIND 75
by the author of Blind 75

Star 79,375 | Tech Interview Handbook | Algorithms Cheatsheet | Changelog | FAQ | About

Indicate your preferences and we will recommend the best LeetCode questions for you to practice.

SCHEDULE
8 weeks
8 hours per week

DIFFICULTY
Easy
Medium
Hard

TOPICS
All topics selected (Change)

Grind 75 questions

Customize LeetCode study plans according to your needs. You are recommended to work on the questions in order.
Find out why.

Questions Summary

TIME NEEDED
How long doing these questions will take for an average person. It's a conservative estimate where it is assumed that roughly the same amount of time will be needed to check the answer for each question.
64 hours
Fits into your schedule.

DIFFICULTY
Questions grouped by difficulty
Easy: 24 Medium: 42 Hard: 9

TOPICS
Questions grouped by topics
Array: 11 Binary: 1 Binary Search: 5 Binary Search Tree: 3 Binary Tree: 9 Dynamic Programming: 5 Graph: 10 Hash Table: 1 Heap: 4 Linked List: 5 Matrix: 1 Recursion: 3 Stack: 7 String: 8 Trie: 2

Need study resources? Check out Tech Interview Handbook's algorithm study cheatsheets.

COMPLETED 0 / 75

You can now bookmark the page to save your preferences! We also recently changed the questions presentation settings. Refer to the changelog for more details.

Stop grinding LeetCode. Study with a plan
Developed by Google engineers, AlgoMonster is the fastest way to get a Software Engineering job. Join today for a 70% discount!

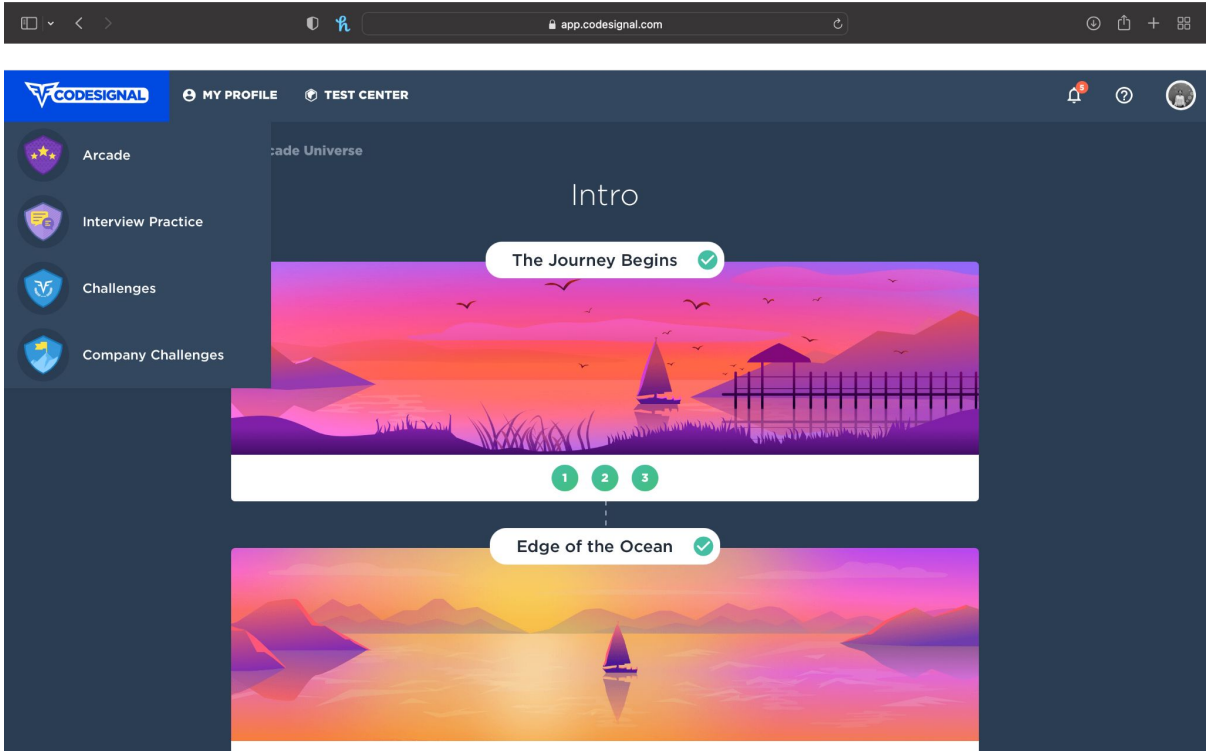
Master Coding Interviews Effectively
Grokking the Coding Interview teaches you techniques and question patterns to be good at coding interviews. Get your lifetime access now!

Craft the Perfect Resume for FAANG
Save time crafting your resume with FAANG Tech Leads' FAANG-quality resume templates and examples which have helped many Software Engineers get interviews at top Bay Area companies. Grab them now for a whopping 70% off!

Free curated interview preparation materials
Tech Interview Handbook goes straight to the point and tells you the minimum you need to know to excel in your technical interviews, and it's free!

PART 02

CodeSignal

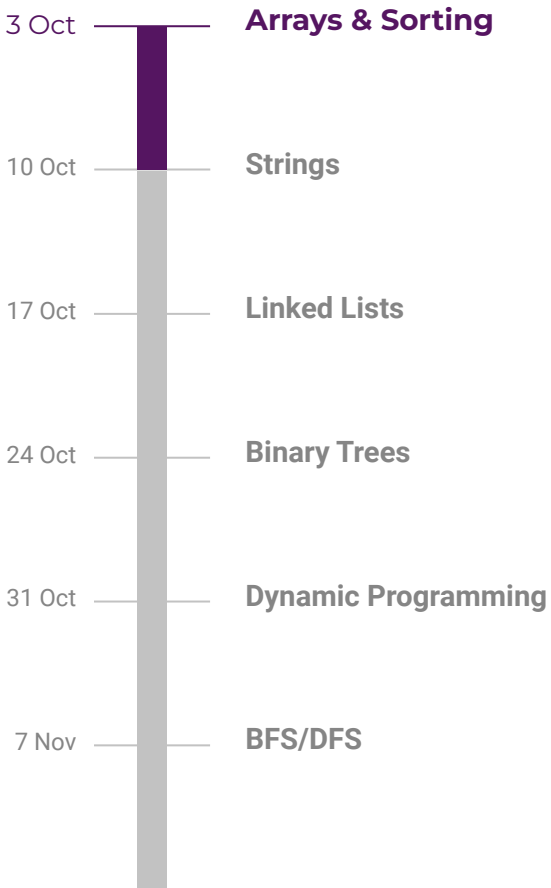


PART 03

Bootcamp Course Structure

PART 03

Bootcamp Timeline



NYU TANDON

Project Brief

Create a website hosted on GitHub for each individual/group. We would use the Open Weather Map API to change the background based on the current location (fetch via browser or IP address) and weather conditions. The UI would include a 24-hour weather forecast and an option for the users to track weather conditions of multiple locations.

For more advanced learners, we could provide a stretch project to add analytical visualizations of historical weather data and predictions.

PART 04

Introduction to Arrays & Sorting

Array

Container of elements of similar data type.

Stored at adjacent memory locations.

Disadvantages

Memory wastage.

Insertion and deletion of elements are costly.

Advantages

Mutable and not fixed in size, making them flexible.

Faster access of elements using the index value.

How to use?

Using the *array* module in Python.

Using the NumPy package arrays.

Built-in Methods

- len( )
- append( )
- clear( )
- count( )
- index( )
- insert( )
- pop( )
- remove( )
- reverse( )
- sort( )

PART 05

Problem Sets

Steps to approach the question:

Understand the problem

Take time to carefully read through the problem from start to finish is critical in finding the correct and complete solution to the problem in hand.

Code your solution

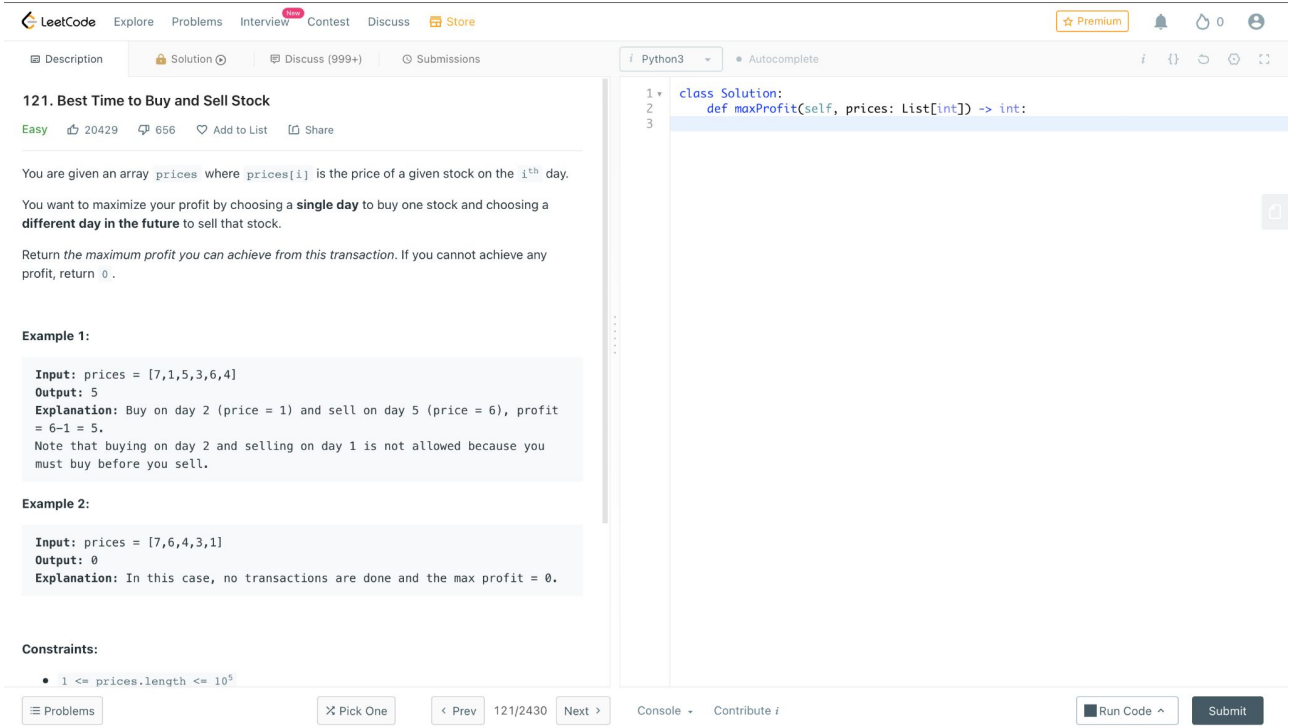
Map out your solution before you write any code. Avoid too much time trying to find the perfect solution. Validate your solution early and often.

Manage your time

Don’t forget, you have multiple questions to complete within a said time. Make sure you allocate enough time to carefully consider all problems.

PART 05

Problem 1: Best time to Buy and Sell Stock



NYU TANDON

Approach 1: Brute Force

```
class Solution:
    def maxProfit(self, prices: List[int]) -> int:
        max_profit = 0
        for i in range(len(prices) - 1):
            for j in range(i + 1, len(prices)):
                profit = prices[j] - prices[i]
                if profit > max_profit:
                    max_profit = profit

        return max_profit
```

Time complexity: $O(n^2)$, loop runs n(n-1)/2 times

Space complexity: O(1), only two variables *max_profit* and *profit*

Approach 2: Single Pass

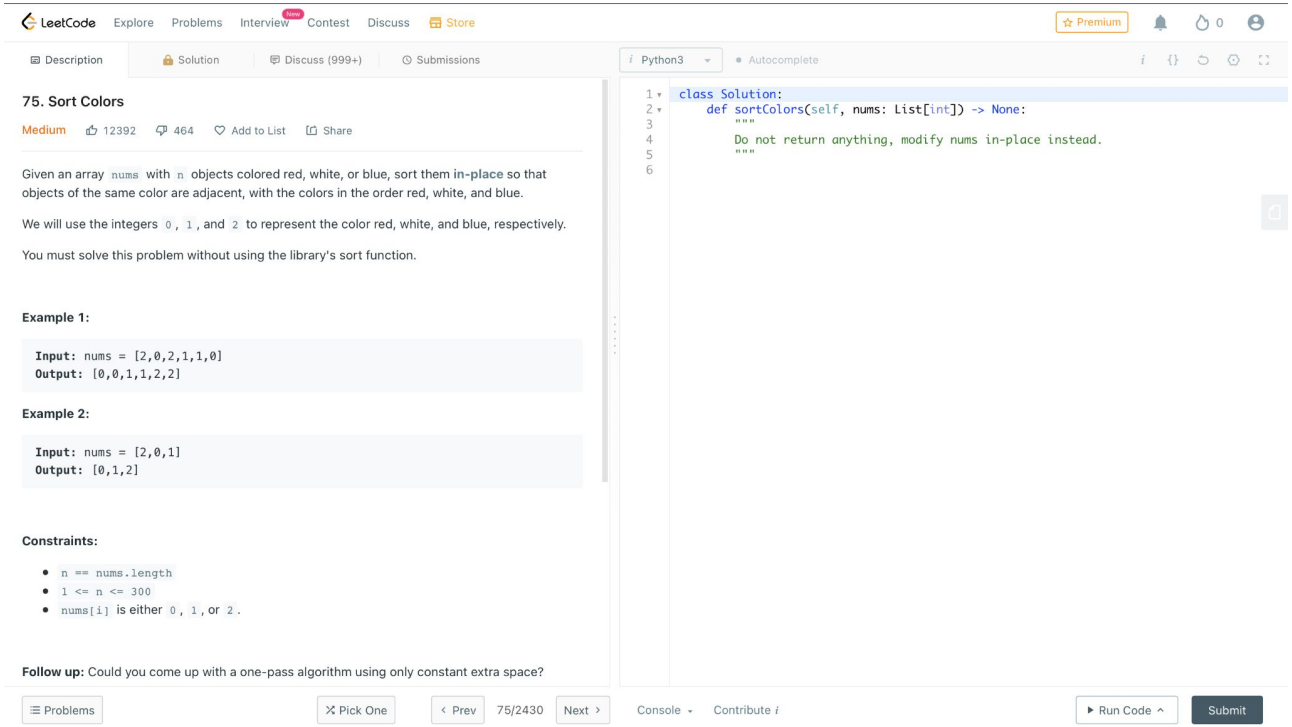
```
class Solution:
    def maxProfit(self, prices: List[int]) -> int:
        min_price = prices[0]
        max_profit = 0
        for price in prices:
            min_price = min(min_price, price)
            max_profit = max(price - min_price, max_profit)

        return max_profit
```

Time complexity: O(n), only single pass is needed

Space complexity: O(1), only two variables *max_profit* and *min_price*

PART 05

Problem 2: Sort Colors



Approach 1: Single Pass

```
class Solution:
    def sortColors(self, nums: List[int]) -> None:
        p0 = curr = 0
        p2 = len(nums) - 1

        while curr <= p2:
            if nums[curr] == 0:
                nums[p0], nums[curr] = nums[curr], nums[p0]
                p0 += 1
                curr += 1
            elif nums[curr] == 2:
                nums[curr], nums[p2] = nums[p2], nums[curr]
                p2 -= 1
            else:
                curr += 1
```

Time complexity: O(n), loop runs one pass along the length

Space complexity: O(1), constant space

PART 06

Q/A

Slack Invite

Join Slack Workspace!

Office Hours: Tuesday (10AM - 2PM)

Problem Assignments

01. Running Sum of 1d Array (Easy)

02. Long Pressed Name (Easy)

03. Contains Duplicate (Easy)

04. Video Stitching (Medium)

05. Maximum Product Subarray (Medium)

06. Container With Most Water (Medium)

07. Sliding Window Maximum (Hard)


NYU TANDON
Thank you!
Upcoming: Strings (10/10)