Back To Course Home

Grokking the Coding Interview: Patterns for Coding Questions

8% completed

Search Course

(/courses/grokking-the-coding-interview/xoyL4q6ApNE)

Problem Challenge 4 (/courses/grokking-the-coding-interview/Y5YDWzqPn7O)

Solution Review: Problem Challenge 4 (/courses/grokking-the-coding-interview/N8nMBvDQJ0m)

Pattern: Two Pointers

Introduction (/courses/grokking-the-coding-interview/xlK78P3XI7E)

Dutch National Flag Problem (medium)

We'll cover the following

- Problem Statement
- Try it yourself
- Solution
  - Code
  - Time complexity
  - Space complexity

Problem Statement

Given an array containing 0s, 1s and 2s, sort the array in-place. You should treat numbers of the array as objects, hence, we can't count 0s, 1s, and 2s to recreate the array.

The flag of the Netherlands consists of three colors: red, white and blue; and since our input array also consists of three different numbers that is why it is called Dutch National Flag problem (https://en.wikipedia.org/wiki/Dutch_national_flag_problem).

Example 1:

Back To Course Home

Grokking the Coding Interview: Patterns for Coding Questions

8% completed

Search Course

(/courses/grokking-the-coding-interview/xoyL4q6ApNE)

Problem Challenge 4 (/courses/grokking-the-coding-interview/Y5YDWzqPn7O)

Solution Review: Problem Challenge 4 (/courses/grokking-the-coding-interview/N8nMBvDQJ0m)

Pattern: Two Pointers

Introduction (/courses/grokking-the-coding-interview/xlK78P3XI7E)

```
Input: [1, 0, 2, 1, 0]
Output: [0 0 1 1 2]
```

Example 2:

```
Input: [2, 2, 0, 1, 2, 0]
Output: [0 0 1 2 2 2 ]
```

Try it yourself

Try solving this question here:

Java | Python3 | JS | C++

```
1  class DutchFlag {
2
3    private static void swap(int[] arr,int i,int j)
4    {
5      int temp = arr[i];
6      arr[i] = arr[j];
7      arr[j] = temp;
8    }
9    public static void sort(int[] arr) {
10     // TODO: Write your code here
11     int n = arr.length;
12     int start = 0, end = n-1;
13     for(int i=0;i<=end;)
14     {
15       if(arr[i]==0)      {
16         swap(arr,start,i);
17         start++;
18         i++;
19       }
20       else if(arr[i]==2){
21         swap(arr,end,i);
22         end--;
```

```
23        }
24        else{
25          i++;
26        }
27      }
28    }
29  }
30
```

Test | Save * | Reset

Show Results | Show Console

2 of 2 Tests Passed

Result	Input	Expected Output	Actual Output	Reason
✓	sort([1, 0, 2, 1, 0])	[0, 0, 1, 1, 2]	[0, 0, 1, 1, 2]	Succeeded
✓	sort([2, 2, 0, 1, 2, 0])	[0, 0, 1, 2, 2, 2]	[0, 0, 1, 2, 2, 2]	Succeeded

3.054s

Solution

The brute force solution will be to use an in-place sorting algorithm like Heapsort (https://en.wikipedia.org/wiki/Heapsort) which will take $O(N * logN)$. Can we do better than this? Is it possible to sort the array in one iteration?

We can use a **Two Pointers** approach while iterating through the array. Let's say the two pointers are called `low` and `high` which are pointing to the first and the last element of the array respectively. So while iterating, we will move all 0s before `low`

Back To Course Home

Grokking the Coding Interview: Patterns for Coding Questions

8% completed

Search Course

(/courses/grokking-the-coding-interview/xoyL4q6ApNE)

Problem Challenge 4 (/courses/grokking-the-coding-interview/Y5YDWzqPn7O)

Solution Review: Problem Challenge 4 (/courses/grokking-the-coding-interview/N8nMBvDQJ0m)

Pattern: Two Pointers

Introduction (/courses/grokking-the-coding-interview/xlK78P3XI7E)

and all 2s after `high` so that in the end, all 1s will be between `low` and `high`.

Code

Here is what our algorithm will look like:

Java | Python3 | C++ | JS

```
class DutchFlag {

  public static void sort(int[] arr) {
    // all elements < low are 0 and all elements >
    // all elements from >= low < i are 1
    int low = 0, high = arr.length - 1;
    for (int i = 0; i <= high;) {
      if (arr[i] == 0) {
        swap(arr, i, low);
        // increment 'i' and 'low'
        i++;
        low++;
      } else if (arr[i] == 1) {
        i++;
      } else { // the case for arr[i] == 2
        swap(arr, i, high);
        // decrement 'high' only, after the swap th
        high--;
      }
    }
  }

  private static void swap(int[] arr, int i, int j)
    int temp = arr[i];
    arr[i] = arr[j];
    arr[j] = temp;
  }

  public static void main(String[] args) {
    int[] arr = new int[] { 1, 0, 2, 1, 0 };
    DutchFlag.sort(arr);
```

Back To Course Home

Grokking the Coding Interview: Patterns for Coding Questions

8% completed

Search Course

(/courses/grokking-the-coding-interview/xoyL4q6ApNE)

Problem Challenge 4 (/courses/grokking-the-coding-interview/Y5YDWzqPn7O)

Solution Review: Problem Challenge 4 (/courses/grokking-the-coding-interview/N8nMBvDQJ0m)

Pattern: Two Pointers

Introduction (/courses/grokking-the-coding-interview/xlK78P3XI7E)

Time complexity

The time complexity of the above algorithm will be $O(N)$ as we are iterating the input array only once.

Space complexity

The algorithm runs in constant space $O(1)$.

Back

Subarrays with Product Less than a Ta...

Next

Problem Challenge 1

Mark as Completed

8% completed, meet the criteria and claim your course certificate!

Claim Certificate

Report an Issue

Ask a Question (https://discuss.educative.io/tag/dutch-national-flag-problem-medium__pattern-two-pointers__grokking-the-coding-interview-patterns-for-coding-questions)

Grokking the Coding Interview: Patterns for Coding Questions

8% completed

Search Course

(/courses/grokking-the-coding-interview/xoyL4q6ApNE)

Problem Challenge 4 (/courses/grokking-the-coding-interview/Y5YDWzqPn7O)

Solution Review: Problem Challenge 4 (/courses/grokking-the-coding-interview/N8nMBvDQJ0m)

Pattern: Two Pointers

Introduction (/courses/grokking-the-coding-interview/xlK78P3XI7E)