Grokking the Coding Interview: Patterns for Coding Questions

74% completed

Search Course

- Introduction
- Single Number (easy)
- Two Single Numbers (medium)
- Complement of Base 10 Number (medium)
- Problem Challenge 1
- Solution Review: Problem Challenge 1

Pattern: Top 'K' Elements

- Introduction
- Top 'K' Numbers (easy)
- Kth Smallest Number (easy)
- 'K' Closest Points to the Origin (easy)
- Connect Ropes (easy)
- Top 'K' Frequent Numbers (medium)
- Frequency Sort (medium)
- Kth Largest Number in a Stream (medium)
- 'K' Closest Numbers (medium)
- Maximum Distinct Elements (medium)
- Sum of Elements (medium)
- Rearrange String (hard)
- Problem Challenge 1
- Solution Review: Problem Challenge 1
- Problem Challenge 2
- Solution Review: Problem Challenge 2
- Problem Challenge 3
- Solution Review: Problem Challenge 3

Pattern: K-way merge

- Introduction
- Merge K Sorted Lists (medium)
- Kth Smallest Number in M Sorted Lists (Medium)
- Kth Smallest Number in a Sorted Matrix (Hard)
- Smallest Number Range (Hard)
- Problem Challenge 1
- Solution Review: Problem Challenge 1

Pattern : 0/1 Knapsack (Dynamic Programming)

- Introduction
- 0/1 Knapsack (medium)
- Equal Subset Sum Partition (medium)
- Subset Sum (medium)
- Minimum Subset Sum Difference (hard)
- Problem Challenge 1
- Solution Review: Problem Challenge 1

Solution Review: Problem Challenge 1

We'll cover the following

- Problem Statement (hard)
  - Code
  - Time Complexity
  - Space Complexity

Problem Statement (hard)

Given a binary matrix representing an image, we want to flip the image horizontally, then invert it.

To flip an image horizontally means that each row of the image is reversed. For example, flipping [0, 1, 1] horizontally results in [1, 1, 0].

To invert an image means that each 0 is replaced by 1, and each 1 is replaced by 0. For example, inverting [1, 1, 0] results in [0, 0, 1].

Example 1:

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

Explanation: First reverse each row: `[[1,0,1],[1,1,1],[1,1,0]]`. Then, invert the image: `[[0,1,0], [0,0,0],[0,0,1]]`

Example 2:

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

Explanation: First reverse each row: `[[0,0,1,1],[1,0,0,1],[1,1,1,0],[0,1,0,1]]`. Then invert the image: `[[1,1,0,0],[0,1,1,0],[0,0,0,1],[1,0,1,0]]`

Solution

- **Flip**: We can flip the image in place by replacing *ith* element from left with the *ith* element from the right.
- **Invert**: We can take XOR of each element with 1. If it is 1 then it will become 0 and if it is 0 then it will become 1.

Code

Here is what our algorithm will look like:

Java | Python3 | C++ | JS

```
function flip_and_invert_image(matrix) {
  const C = matrix.length;
  for (var row = 0; row < C; ++row) {
    for (var col = 0; col < Math.floor((C + 1) / 2); ++col) {
      var tmp = matrix[row][col] ^ 1;
      matrix[row][col] = matrix[row][C - 1 - col] ^ 1;
      matrix[row][C - 1 - col] = tmp;
    }
  }
  return matrix;
}

console.log(flip_and_invert_image([[1,0,1], [1,1,1], [0,1,1]]))
console.log(flip_and_invert_image([[1,1,0,0],[1,0,0,1],[0,1,1,1],[1,0,1,0]]))
```

RUN SAVE RESET

Problem Challenge 2
Solution Review: Problem Challenge 2

Pattern: Topological Sort (Graph)

Introduction
Topological Sort (medium)
Tasks Scheduling (medium)
Tasks Scheduling Order (medium)
All Tasks Scheduling Orders (hard)
Alien Dictionary (hard)
Problem Challenge 1
Solution Review: Problem Challenge 1
Problem Challenge 2
Solution Review: Problem Challenge 2

Miscellaneous

Kth Smallest Number (hard)

Conclusions

Where to Go from Here

Close

Output 2.141s

```
[ [ 0, 1, 0 ], [ 0, 0, 0 ], [ 0, 0, 1 ] ]
[ [ 1, 1, 0, 0 ], [ 0, 1, 1, 0 ], [ 0, 0, 0, 1 ], [ 1, 0, 1, 0 ] ]
```

Time Complexity

The time complexity of this solution is $O(n)$ as we iterate through all elements of the input.

Space Complexity

The space complexity of this solution is $O(1)$.