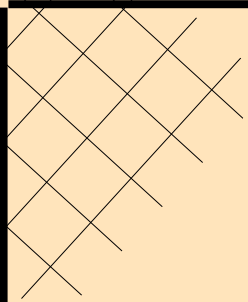


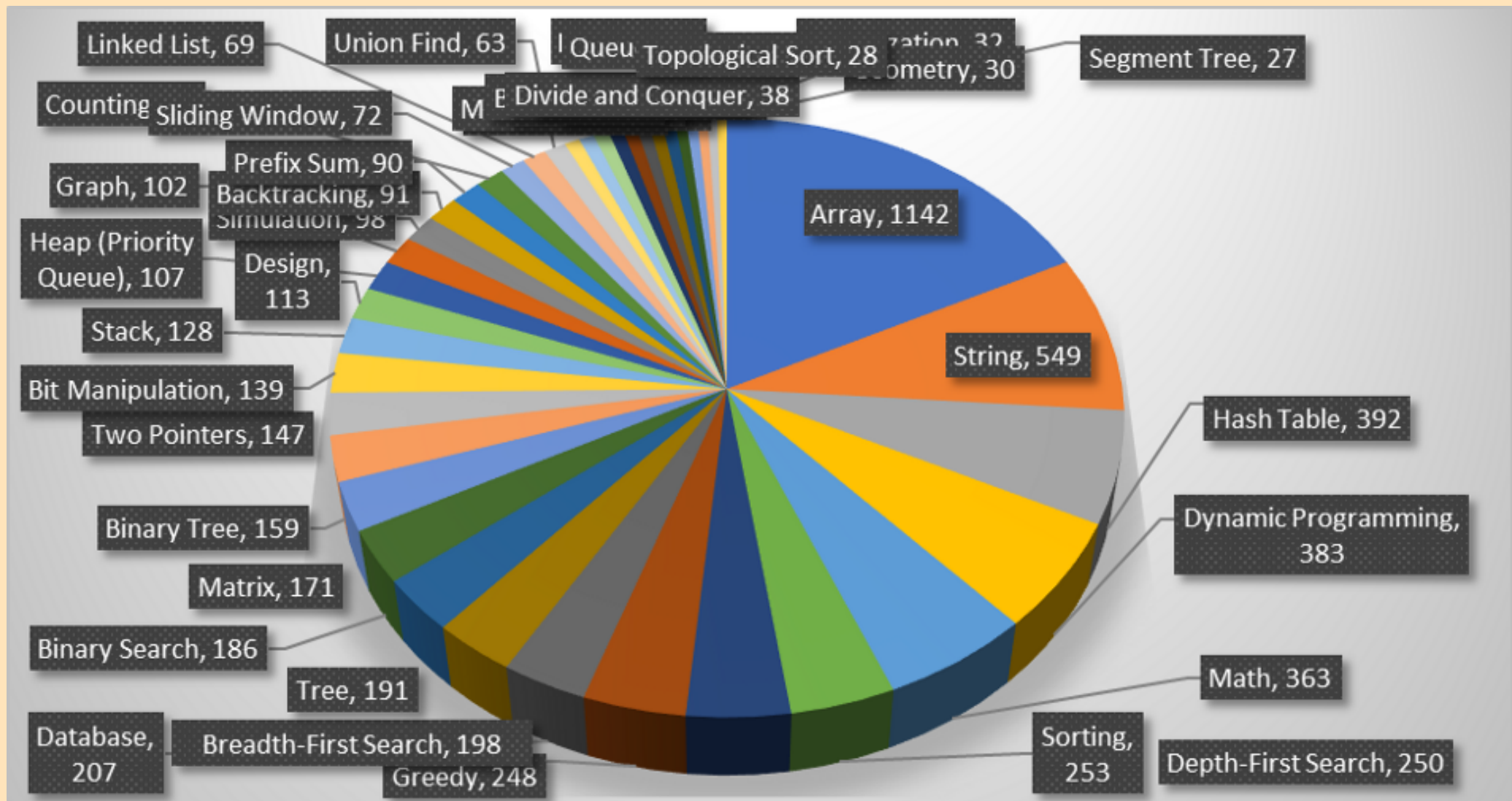
# **Top LeetCode Coding Patterns with Highest ROI**

**DesignGurus.org**

- 
- ➡ LeetCode (LC), being the largest repository of coding problems, contains more than **2k+** questions.
  - ➡ What types of coding questions one should **focus** on?
  - ➡ Which algorithmic techniques have the **highest ROI**?

# LeetCode Topic Distribution

Here is the topic distribution for LC questions:





# Top Data Structures with Best ROI

Here are the top Data Structures with the highest return on investment:

1. **Array** (1142 problems)
2. **String** (549)
3. **Hash Table** (392)
4. **Tree** (191)
5. **Matrix** (171)
6. **Stack** (128)
7. **Heap** or **Priority Queue** (107)
8. **Graph** (102)
9. **Linked** List (69)
10. **Trie** (44)

# Top Algorithmic Techniques with Best ROI

Here are the top algorithmic techniques with the highest return on investment:

1. **Dynamic Programming** (383)
2. **Sorting** (253)
3. **Greedy** (248)
4. **Binary Search** (186)
5. **Backtracking** (91)
6. **Recursion** (44)
7. **Divide and Conquer** (38)



# Top Coding Patterns with Best ROI

Here are the top coding patterns with the highest return on investment:

1. **Depth First Search** (250)
2. **Breadth First Search** (198)
3. **Binary Search** (186)
4. **Two Pointers** (147)
5. **Sliding Window** (72)
6. **Monotonic Stack** (44)
7. **Union Find** (63)
8. **Memoization** (32)
9. **Topological Sort** (28)
10. **Segment Tree** (27)



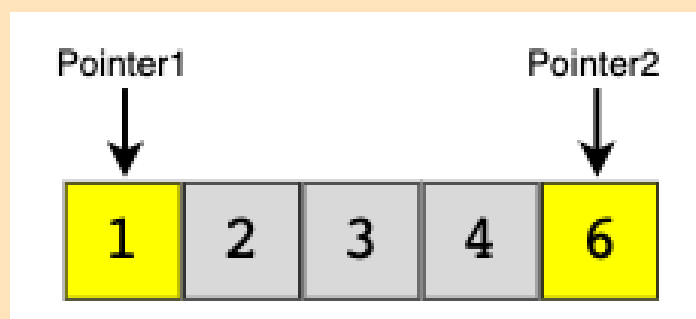
## **7 Best Coding Patterns with Highest ROI**

Combining all categories from the above data, here is the list of best coding patterns/techniques with the highest ROI.

# 1. Two Pointers

**LC Tags:** Array, String, Fast & Slow Pointer

This pattern covers a huge set of questions related to Arrays and Strings, which are the highest tagged data structures. Fast & Slow Pointer can be understood easily as it is a variation of the Two Pointers pattern.

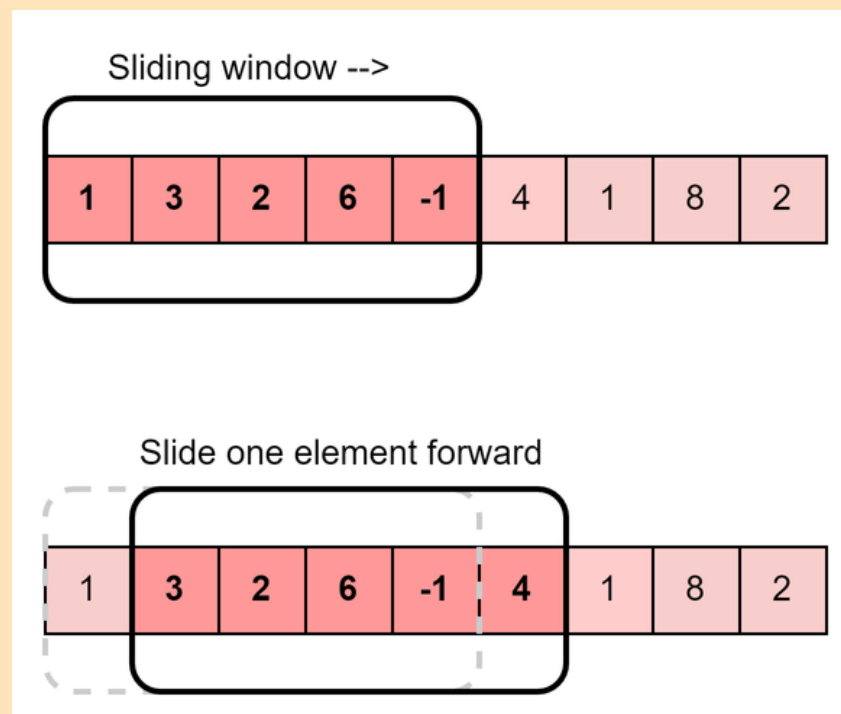




## 2. Sliding Window

**LC Tags:** Arrays, Strings, Hash Tables

Sliding Window covers most of the problems related to top data structures like Arrays, Strings, and HashTables.



### 3. Tree/Graph Breadth-First Search

**LC Tags:** Tree, Graph, Queue, Subsets, Matrix Traversal, Topological Sort

Breadth First Search (BFS) is a very handy pattern. BFS's patterns like Subsets, Matrix Traversal, and Topological Sort cover many problems.

## 4. Tree/Graph Depth First Search

**LC Tags:** Tree, Graph, Matrix Traversal

Most Trees and Graphs problems can be solved using Depth First Search (DFS). Matrix Traversal, which is also DFS based pattern, covers most of the matrix-related problems.



## 5. Modified Binary Search

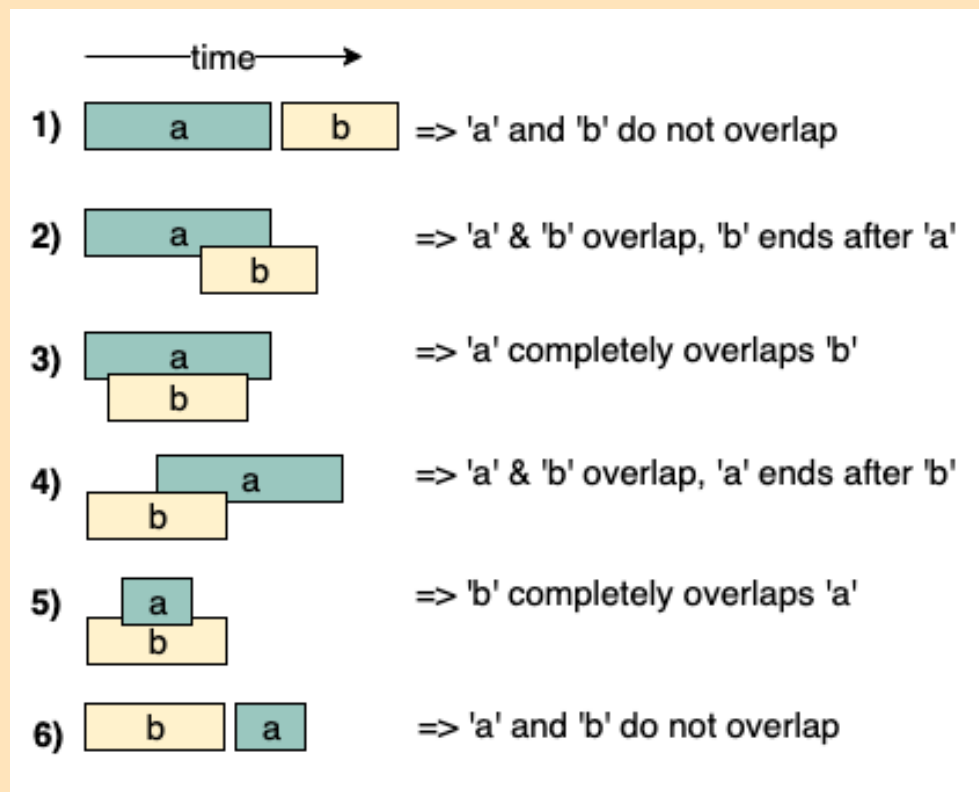
**LC Tags:** Array, Binary Search

Binary Search and its variants are used to solve a huge number of coding questions.

## 6. Merge Intervals

**LC Tags:** Array, Heap

Although there are not many problems related to Interval Merge, these problems frequently appear in coding interviews

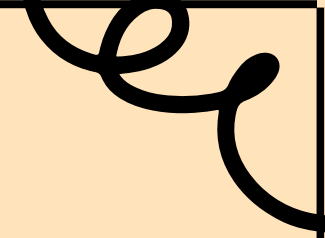




## 7. Recursion/Backtracking

**LC Tags::** Array, Queue, Matrix

Backtracking and recursion are used to solve a wide range of problems. Mastering these techniques is highly recommended.



➡ Coding patterns enhance our “**ability to map a new problem to an already known problem.**”

➡ Follow these techniques to distinguish yourself from others! A number of these approaches are discussed in “**Grokking the Coding Interview**” and “**Grokking Dynamic Programming.**”