

ARRAYS – 100% COMPLETE TOPIC & PATTERN CHECKLIST

1. ARRAY BASICS (FOUNDATION – MUST KNOW)

Core Concepts

- What is an array (contiguous memory)
 - Indexing (0-based vs 1-based)
 - Static vs dynamic arrays
 - 1D vs 2D arrays
 - In-place operations
 - Pass array to function
 - Time & space complexity of array operations
-

2. BASIC ARRAY OPERATIONS

◆ Patterns

- **Linear traversal**
- **Accumulator pattern**

Topics

- Traversing array
 - Inserting an element
 - Deleting an element
 - Updating elements
 - Finding max / min
 - Second largest / smallest
 - Sum / product of elements
 - Count even / odd
 - Frequency count (without hashing)
-

3. REVERSAL & ROTATION (VERY COMMON)

◆ Patterns

- Two pointers
- Reversal algorithm

✅ Topics

- Reverse array
 - Rotate left / right
 - Rotate by K positions
 - Cyclic rotation
 - Check if array is rotated
 - Juggling algorithm
-

4. PREFIX SUM & DIFFERENCE ARRAY

◆ Patterns

- Prefix sum
- Range query optimization

✅ Topics

- Prefix sum array
 - Range sum queries
 - Subarray sum using prefix sum
 - Difference array
 - Range update queries
 - Equilibrium index
-

5. SUBARRAY PROBLEMS (VERY HIGH WEIGHTAGE)

◆ Patterns

- Carry forward

- **Sliding window**
- **Kadane's algorithm**

✓ Topics

- Generate all subarrays
 - Maximum subarray sum
 - Minimum subarray sum
 - Subarray with given sum
 - Count subarrays with sum = K
 - Longest subarray with sum K
 - Zero sum subarray
 - Subarray product problems
-

6. SLIDING WINDOW (CORE INTERVIEW PATTERN)

◆ Patterns

- **Fixed window**
- **Variable window (expand–shrink)**

✓ Topics

- Maximum sum subarray of size K
 - Minimum sum window
 - Longest subarray with conditions
 - Window max / min
 - Sliding window optimization ($O(n)$)
-

7. TWO POINTER TECHNIQUE

◆ Patterns

- **Opposite ends**
- **Same direction**

✓ Topics

- Pair sum (sorted / unsorted)
 - Remove duplicates (sorted)
 - Merge two sorted arrays
 - Move all zeros to end
 - Segregate even/odd
 - Dutch National Flag (0s,1s,2s)
 - Container with most water
-

🔍 8. BINARY SEARCH ON ARRAYS

◆ Patterns

- Search space reduction
- Binary search on answer

✓ Topics

- Binary search (iterative + recursive)
 - First & last occurrence
 - Count occurrences
 - Lower bound / upper bound
 - Search in rotated sorted array
 - Peak element
 - Minimum in rotated array
 - Square root using BS
-

🔄 9. SORTING + ARRAY LOGIC

◆ Patterns

- Sort + scan
- Two pointer after sort

✓ Topics

- Bubble / Selection / Insertion (basics)
 - Merge sort (important)
 - Quick sort (important)
 - Sort colors (0,1,2)
 - Inversion count
 - Kth largest / smallest
 - Merge intervals
 - Meeting rooms
-

10. HASHING WITH ARRAYS (COMBINATION)

◆ Patterns

- **Frequency map**
- **Prefix sum + hashing**

✓ Topics

- Two sum
 - Majority element ($> n/2$)
 - Missing number
 - Repeating number
 - Longest consecutive sequence
 - Count distinct elements
 - Subarray with sum = K
-

11. MATRIX / 2D ARRAYS (MUST FOR TESTS)

◆ Patterns

- **Row-column traversal**
- **Layer-by-layer**

✓ Topics

- Matrix input/output
 - Row-wise & column-wise traversal
 - Spiral traversal
 - Transpose matrix
 - Rotate matrix (90°)
 - Search in 2D matrix
 - Diagonal traversal
 - Matrix prefix sum
-

🧠 12. ADVANCED ARRAY PATTERNS (INTERVIEW DECIDERS)

◆ Patterns

- Greedy + array
- Binary search on answer
- Mathematical observation

✓ Topics

- Trapping rain water
 - Maximum product subarray
 - Best time to buy & sell stock (all variants)
 - Majority element ($n/3$)
 - Rearranging array
 - Product of array except self
 - Minimum jumps to reach end
-

⚠ 13. EDGE CASES & TRICKS (VERY IMPORTANT)

✓ Must Handle

- Empty array

- Single element
 - All elements same
 - Negative numbers
 - Large constraints
 - Overflow cases
 - In-place vs extra space
-

❏ FINAL ARRAY CHECKPOINT

You are **array-ready** if you can:

- Identify **pattern in 30 seconds**
- Solve most problems in **$O(n)$ or $O(n \log n)$**
- Explain brute force → optimized approach
- Handle edge cases confidently