AI Assistance Documentation - prompts.docx

# Q1: Find Missing Numbers in Array

\*\*Prompt Used:\*\*  
Copilot: Generate function to find missing numbers from array where numbers range from 1 to n

\*\*Response Received:\*\*  
Copilot suggested using a for-loop to mark indices using negative values and collect indices with positive values as the missing numbers.

\*\*Implementation Details:\*\*  
Used the suggestion as-is with Math.Abs to safely mark the indices and ensure duplicate values don't cause index errors.

\*\*Adjustments:\*\*  
No major changes. Added comments and validated edge cases like all numbers present.

# Q2: Sort Array by Parity

\*\*Prompt Used:\*\*  
Copilot: Write a function to sort an array by moving even numbers to the front and odd numbers to the back

\*\*Response Received:\*\*  
Copilot suggested a two-pointer approach swapping values when an even-odd mismatch was found.

\*\*Implementation Details:\*\*  
Used Copilot’s code mostly unchanged. Ensured the return value is in-place.

\*\*Adjustments:\*\*  
No changes needed. Included edge case for all even or all odd inputs.

# Q3: Two Sum

\*\*Prompt Used:\*\*  
Copilot: Find two indices in an array whose values sum up to a given target

\*\*Response Received:\*\*  
Copilot suggested using a Dictionary to track previously seen values and their indices.

\*\*Implementation Details:\*\*  
Implemented using Copilot’s logic. Efficient lookup and return when a complement is found.

\*\*Adjustments:\*\*  
Minor renaming for readability.

# Q4: Maximum Product of Three Numbers

\*\*Prompt Used:\*\*  
Copilot: Find the maximum product of any three numbers in an array

\*\*Response Received:\*\*  
Suggested sorting the array and comparing product of three largest vs two smallest and the largest.

\*\*Implementation Details:\*\*  
Used the same logic as suggested. Handled negative numbers as well.

\*\*Adjustments:\*\*  
None.

# Q5: Decimal to Binary Conversion

\*\*Prompt Used:\*\*  
Copilot: Convert a given decimal number to binary string representation

\*\*Response Received:\*\*  
Suggested using while loop and modulo to build binary string from right to left.

\*\*Implementation Details:\*\*  
Implemented the logic exactly. Covered edge case for 0 input.

\*\*Adjustments:\*\*  
No changes.

# Q6: Find Minimum in Rotated Sorted Array

\*\*Prompt Used:\*\*  
Copilot: Find the minimum element in a rotated sorted array

\*\*Response Received:\*\*  
Suggested binary search by comparing middle and rightmost values.

\*\*Implementation Details:\*\*  
Used suggestion with proper mid-calculation to avoid overflow.

\*\*Adjustments:\*\*  
None.

# Q7: Palindrome Number

\*\*Prompt Used:\*\*  
Copilot: Check if a given integer is a palindrome number

\*\*Response Received:\*\*  
Suggested reversing the number and comparing with the original.

\*\*Implementation Details:\*\*  
Used Copilot’s logic. Edge case handled for negative numbers.

\*\*Adjustments:\*\*  
None.

# Q8: Fibonacci Number

\*\*Prompt Used:\*\*  
Copilot: Write a function to return the nth Fibonacci number

\*\*Response Received:\*\*  
Suggested using iterative approach with two variables.

\*\*Implementation Details:\*\*  
Used exactly as suggested. Efficient and passes all tests.

\*\*Adjustments:\*\*  
None.