**ISM 6225: Application Development For Analytics  
Assignment 2- Computational Problem Solving   
Done BY: Neethika Kommidi  
Date of Submission: 04-06-2025**

**Step-1: Clone the starter repository**

* First, I cloned the GitHub repository using GitHub
* Copied the clone URL on GitHub
* Used GitHub Desktop → File → Clone Repository
* Opened Visual Studio solution

This gave me access to the correct starter code (Program.cs) with all method templates.

**Step-2: Implement Algorithms**

In Visual Studio:

* I wrote in the Program.cs file.
* For each method, I read the description and added a natural language comment.
* GitHub Copilot provided a code suggestion.

I verified each function by running the sample inputs from the Main() method.

**Question-1: Finding missing numbers in array**

**Problem Statement**:

Find numbers between 1 and n missing from the given array.

**Approach:**

In-place marking using index negation to mark visited values.

**Copilot Prompt & Suggestion:**

Asked Copilot to say "find all numbers missing in range using in-place index method."

It suggested marking elements negative at respective indices and taking positives.

**Concepts Learned:**

Array indexing

Negation trick to save extra space

Edge Cases:

All numbers present

Duplicates

Empty array

**Question -2: Sort array by parity**

**Problem Statement:**

Reverse the array such that even numbers precede odd numbers.

**Approach:**

Two-pointer solution to swap even and odd values efficiently.

**Copilot Prompt & Suggestion:**

I wrote "place evens before odds using two-pointer logic."

Copilot responded with a loop exchanging values from both ends.

**Concepts Learned:**

In-place sorting

Pointer traversal

**Edge Cases:**

Only odd/even numbers

Empty array

Alternating pattern

**Question 3: Two Sum**

**Problem Statement:**

Find two indices whose sum is equal to a target.

**Solution:**

Used a hash map to store values and query complements.

**Copilot Prompt & Suggestion:**

I asked Copilot to "return indices of two numbers that sum to target using hash table."

It used a dictionary to query complement and maintain indices.

**Concepts Learned:**

Dictionary lookup

Time complexity reduction

**Edge Cases:**

Duplicate numbers

Negative values

No valid pairs

**Question 4: Maximum Product of Three Numbers**

**Problem Statement:**

Find the maximum product achievable using any 3 numbers from the array.

**Approach:**

Sort the array and consider two cases: product of 3 largest and 2 smallest + largest.

**Copilot Prompt & Suggestion:**

Given prompt "calculate highest product of three using sorted values."

Copilot suggested comparing values using sorted array indices.

**Concepts Learned:**

Negative values

Array sorting

**Edge Cases:**

Negative products

Zeros

Repeated values

**Question 5: Decimal to Binary Conversion**

**Problem Statement:**

Convert a decimal number to its binary string representation.

**Solution:**

Used repeated division by 2 and built up remainders.

**Copilot Prompt & Suggestion:**

Prompt: "Decimal to binary through division method."

Suggestion: Remainder collection loop with string reversal.

**Concepts Learned:**

Base conversion

Manual binary calculation

**Edge Cases:**

Input is 0

Big numbers

**Question 6: Find Minimum in Rotated Sorted Array**

**Problem Statement:**

Return the minimum value in a rotated sorted array.

**Solution:**

Used modified binary search to find inflection point.

**Copilot Prompt & Suggestion:**

Prompted with: "Find smallest element in rotated sorted array using binary search."

Copilot prompted with mid-point comparison with right element.

**Concepts Learned:**

Binary search

Searching sorted circular data

**Edge Cases:**

Already sorted

Rotation at different positions

**Question 7: Palindrome Number**

**Problem Statement:**

Check whether given integer is a palindrome.

**Approach:**

Mathematically reversed number compared with original.

**Copilot Prompt & Suggestion**

Instruction: "Check if number is palindrome without converting to string."

Copilot used modulo and division to reverse digits.

**Concepts Learned:**

Reversal of number

Prevention of string methods

**Edge Cases:**

Negative input

Zero-ending numbers

**Question 8 : Fibonacci Number**

**Problem Statement:**

Return the nth Fibonacci number.

**Approach:**

Applied iterative calculation to avoid recursion stack.

**Copilot Prompt & Suggestion:**

Prompted with "return nth Fibonacci number using loop."

Copilot suggested keeping two previous numbers and looping.

**Concepts Learned:**

Dynamic programming

Iterative optimization

**Edge Cases:**

n = 0 or 1

Large inputs

**Microsoft Visual Studio Debug Console:**

