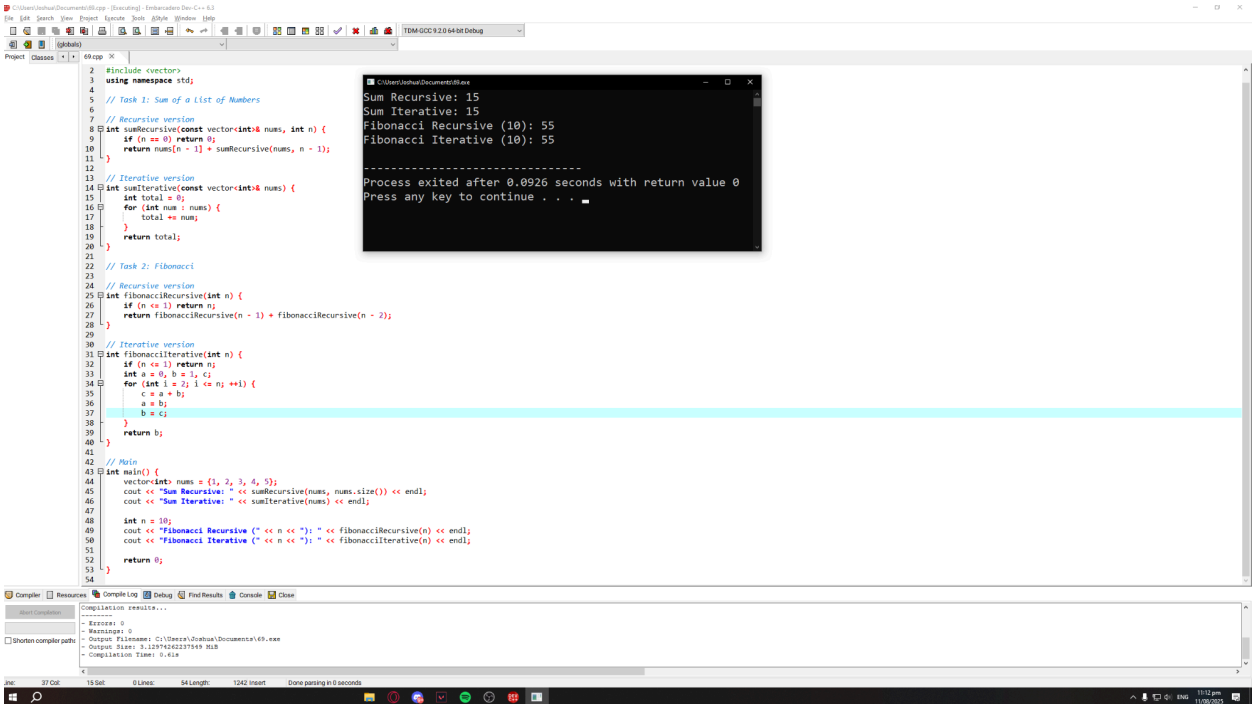
Name: Alcantara, Jason P. 8/12/25
CPE 010 - CPE21S4 Engr. Jimlord Quejado



Explanation:
The Big-O-Notation shows how quick and how much memory an algorithm uses as the input gets larger, focusing on the worst-case-scenario.

Task 1: Summing a List of Numbers

- **In the recursive version, the time complexity is O(n) because the function runs once for each item in the list, But the space complexity is 0(1) because it uses only a fixed amount of memory.**

Task 2: Fibonacci Sequence

- **The recursive version, the time complexity is O(2^n) because each number calls the function twice.**
- **The non recursive version or Iterative version since it loops through numbers once, and the space complexity is O(1) as it only uses one or two variables.**