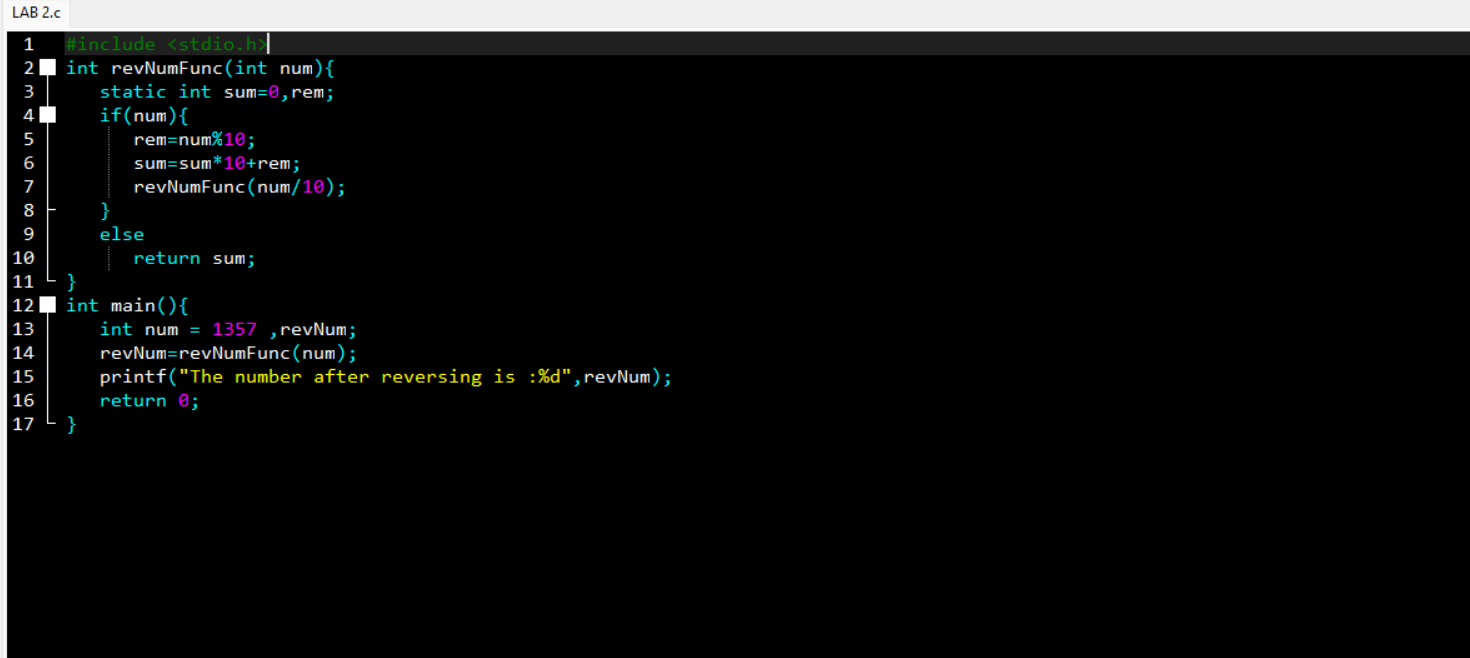
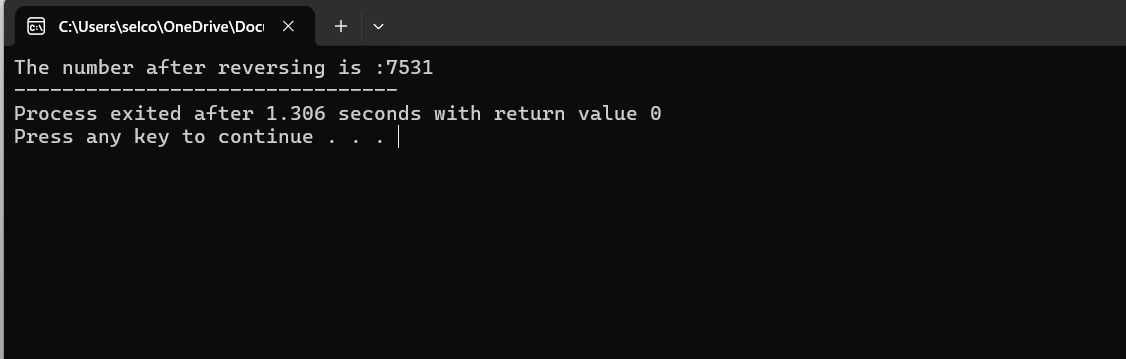
LAB ASSESSMENT 2

1)REVERSE A GIVEN NUMBER IN C

CODE:

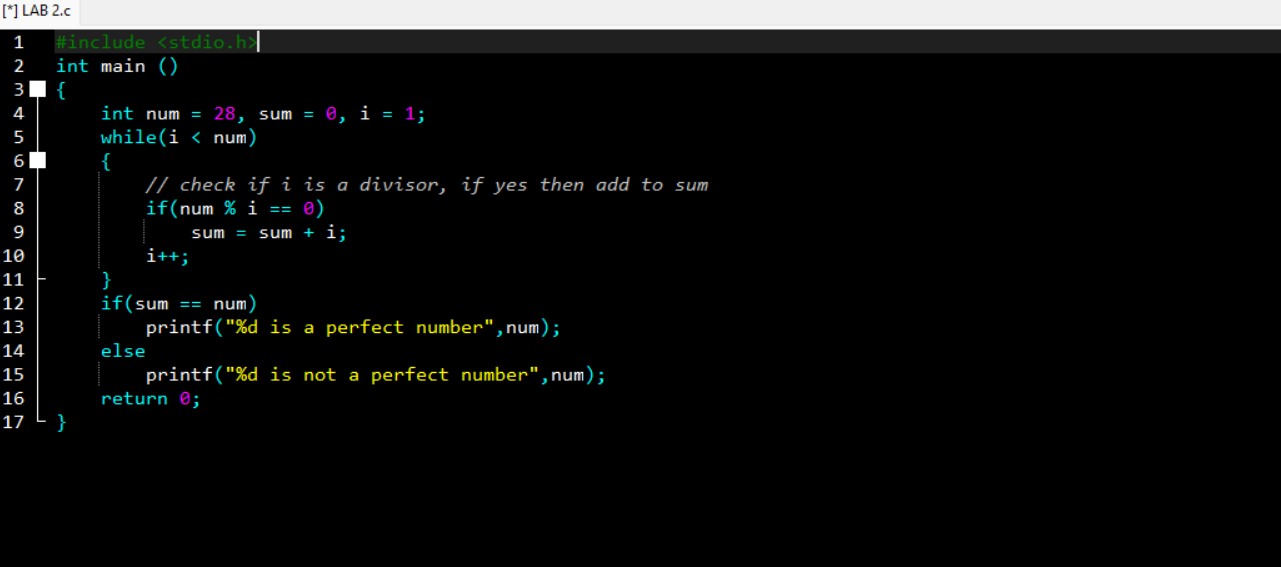


OUTPUT:

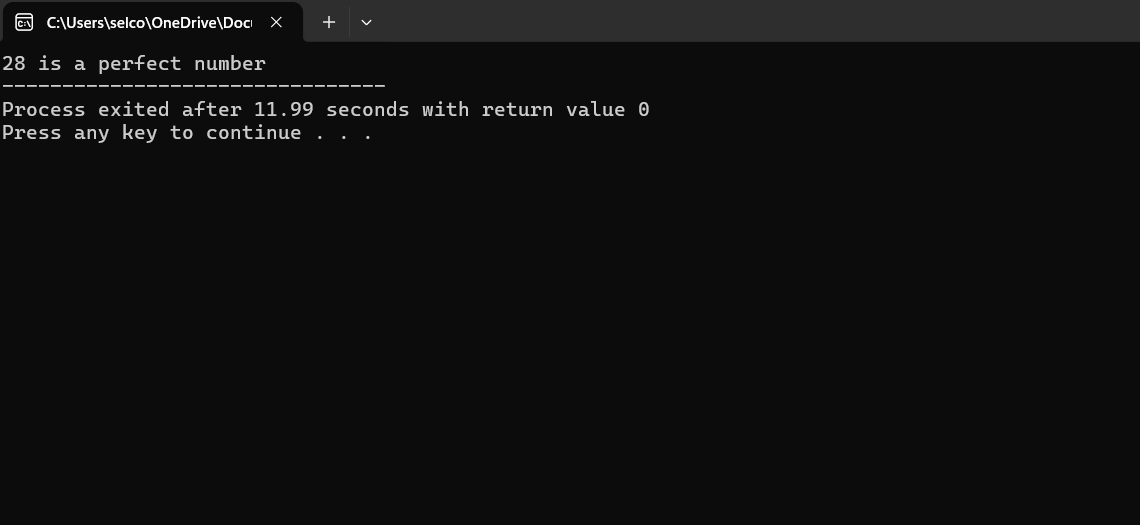


2)FINDING A PERFECT NUMBER:

CODE:

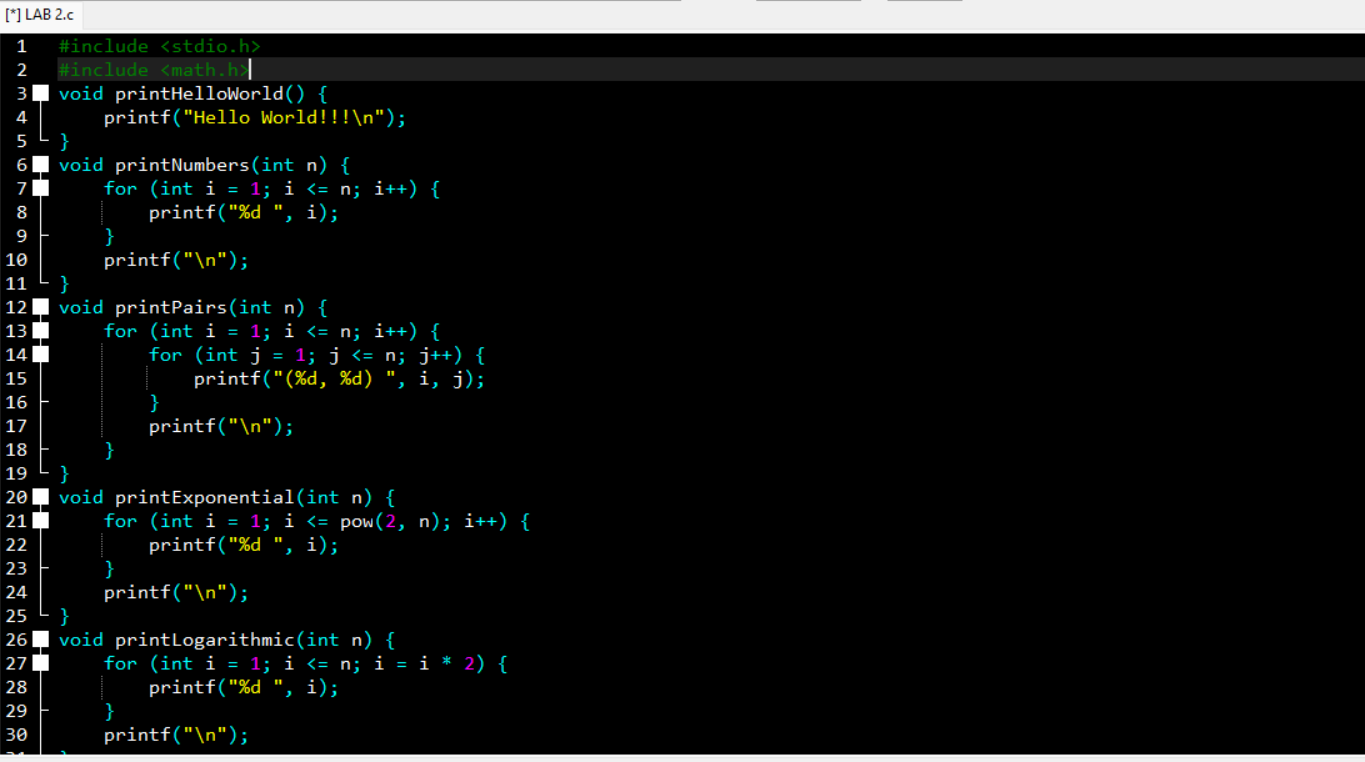


OUTPUT:



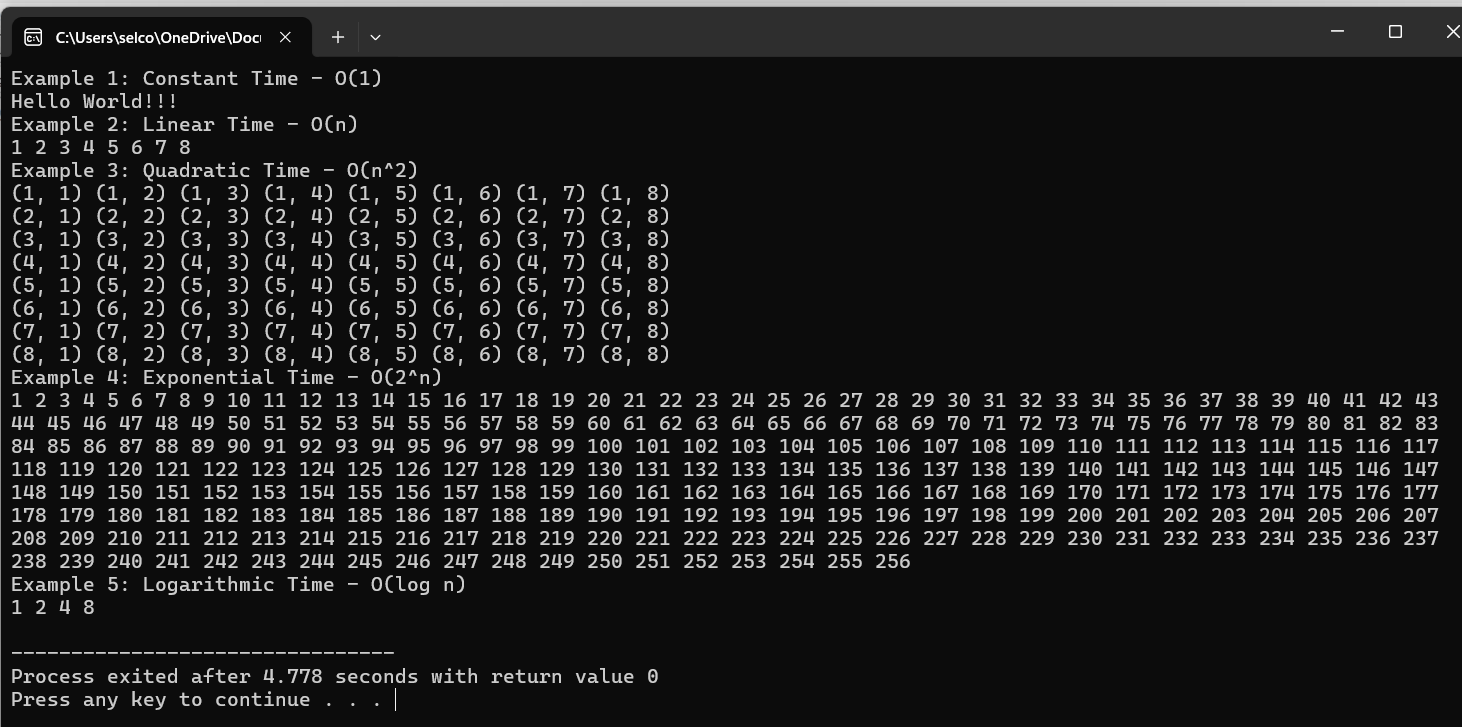
3)USAGE OF NOTATIONS USING TIME COMPLEXITY OF SOME CODES

CODE:





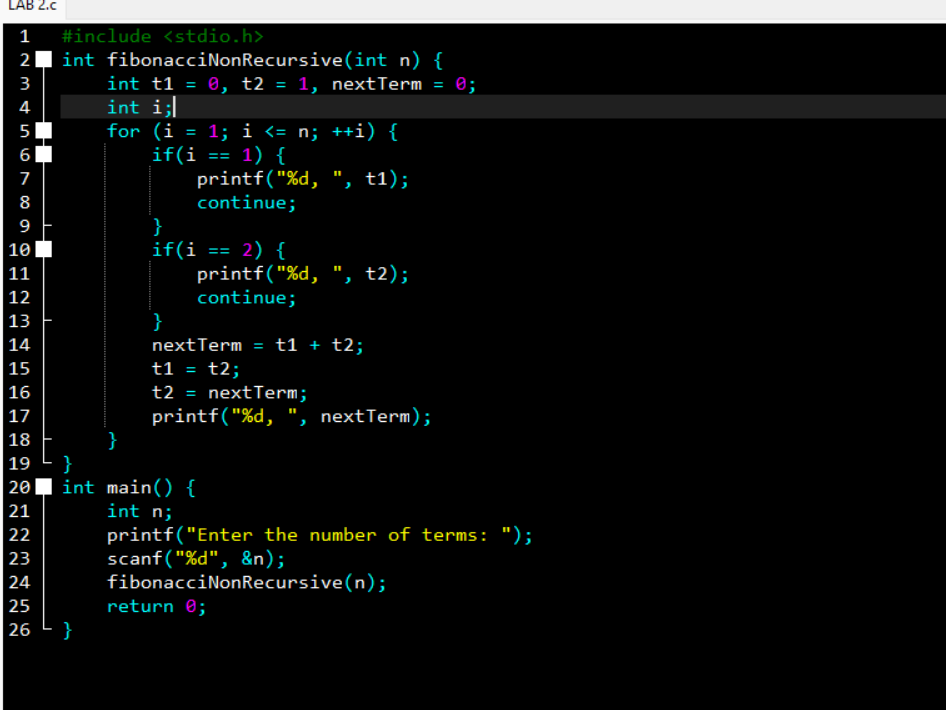
OUTPUT:



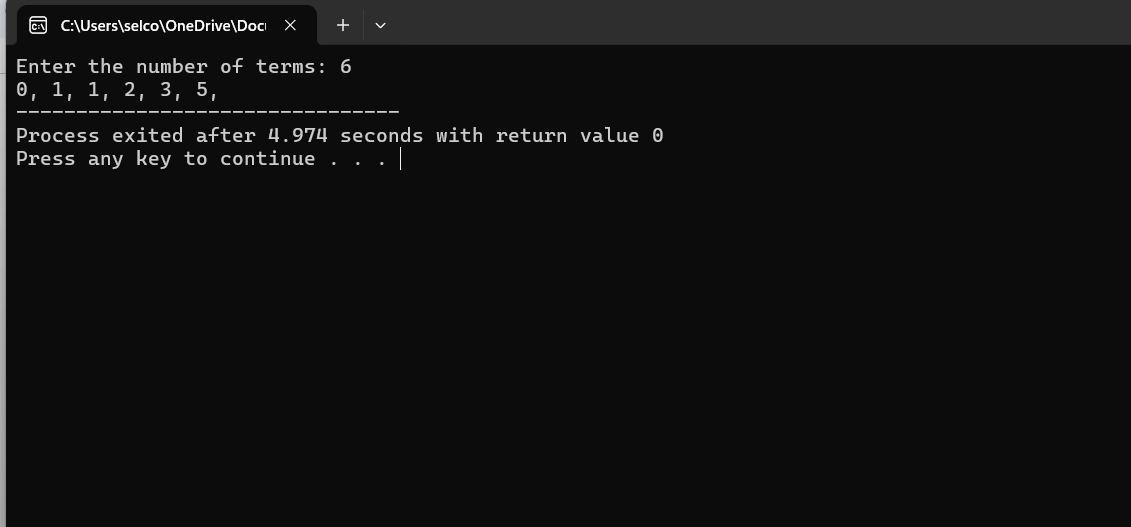
4)DEMONSTRATE THE MATHEMATICAL ANALYSIS OF RECURSIVE AND NON RECURSIVE ALGORITHMS

CODE:

NON RECURSIVE ALGORITHM:FIBANACCI SERIES

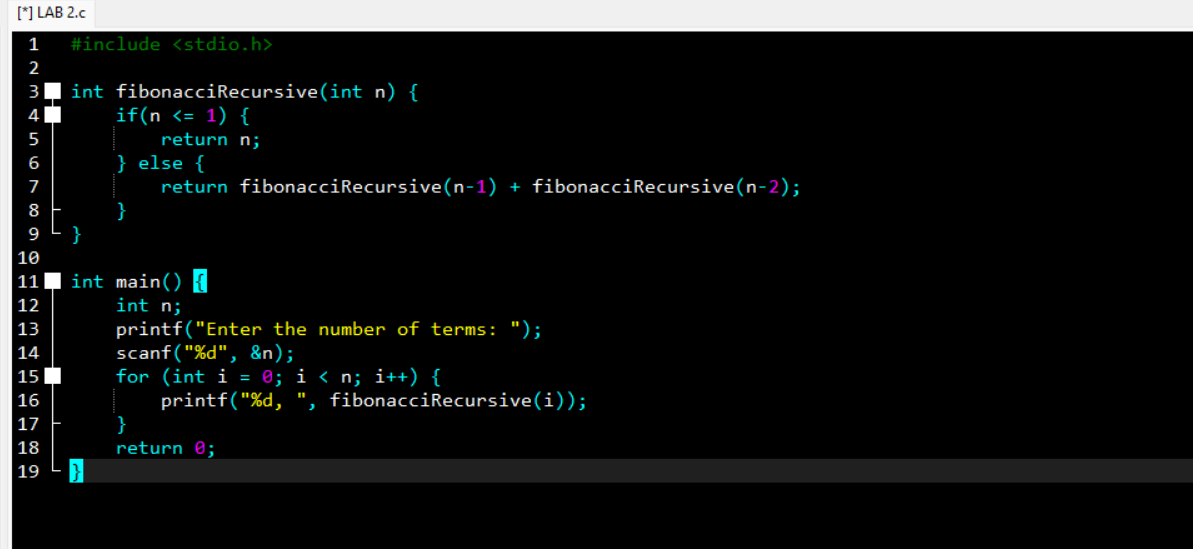


Output:

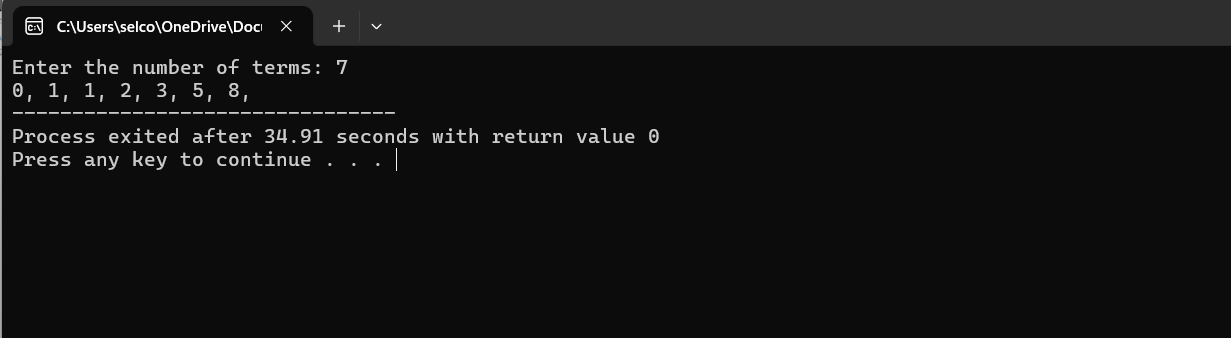


RECURSIVE ALGORITHM :FIBANACCI SERIES

CODE:

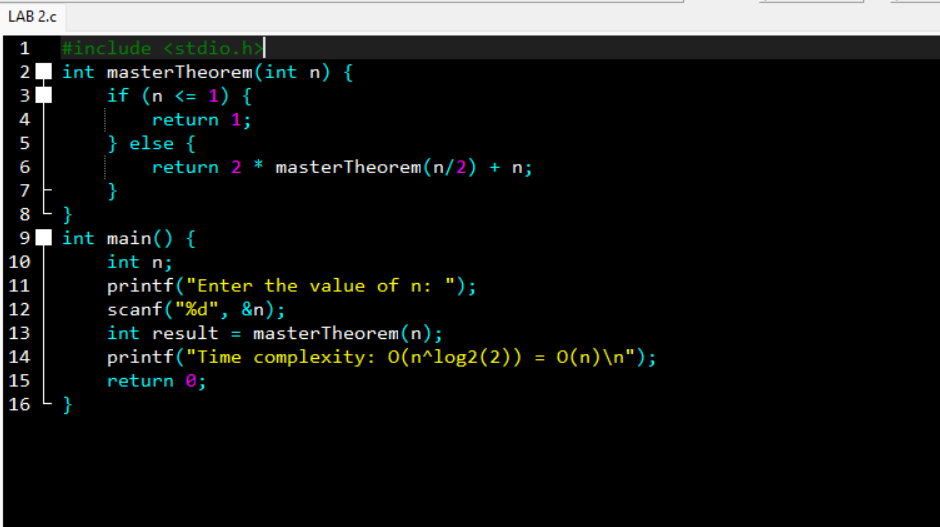


OUTPUT:

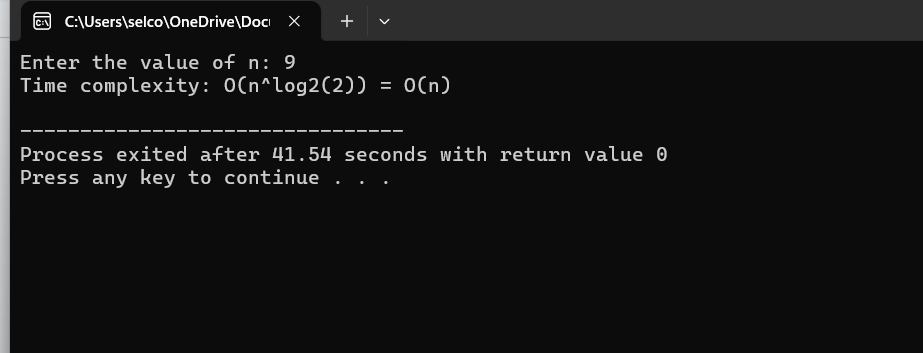


5)solving recurrence relations using the Master Theorem, Substitution Method, and Iteration Method will demonstrate how to calculate the time complexity of an example recurrence relation using the specified technique

CODE:

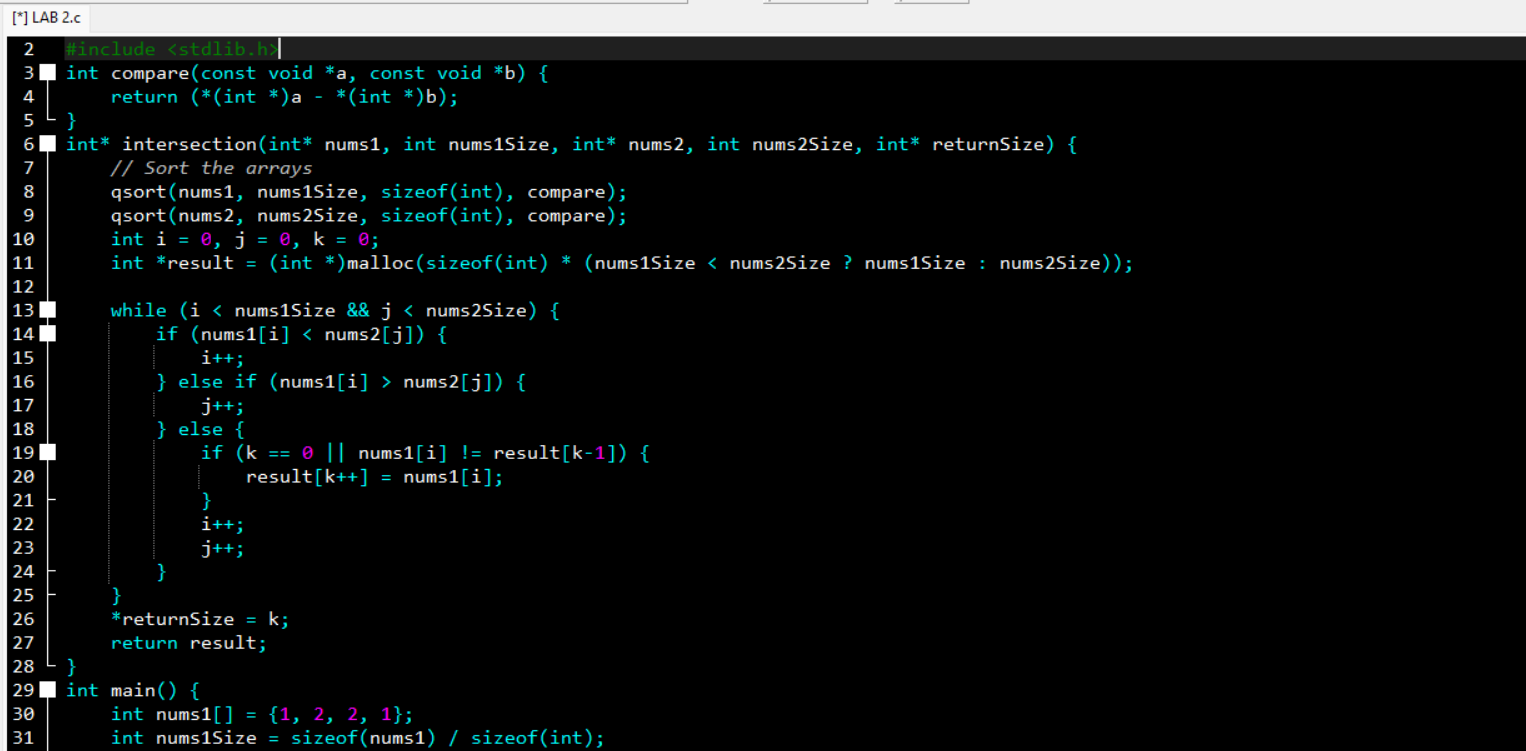


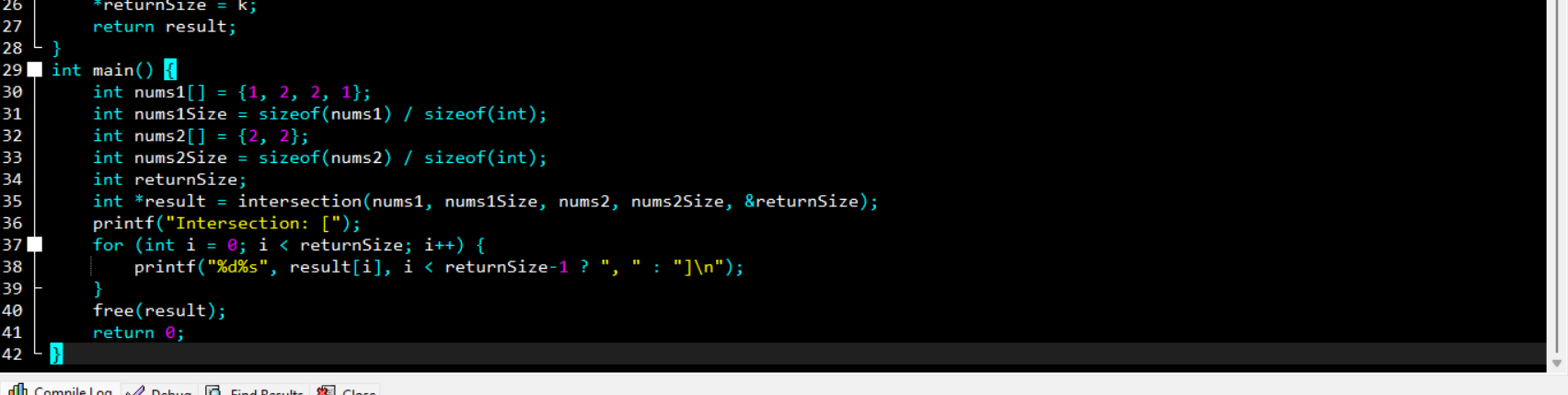
OUTPUT:



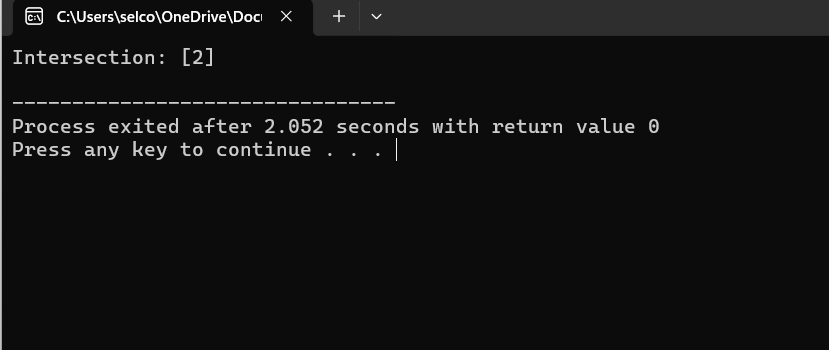
6)Given two integer arrays nums1 and nums2, return an array of their Intersection. Each element in the result must be unique and you may return the result in any order.

CODE:



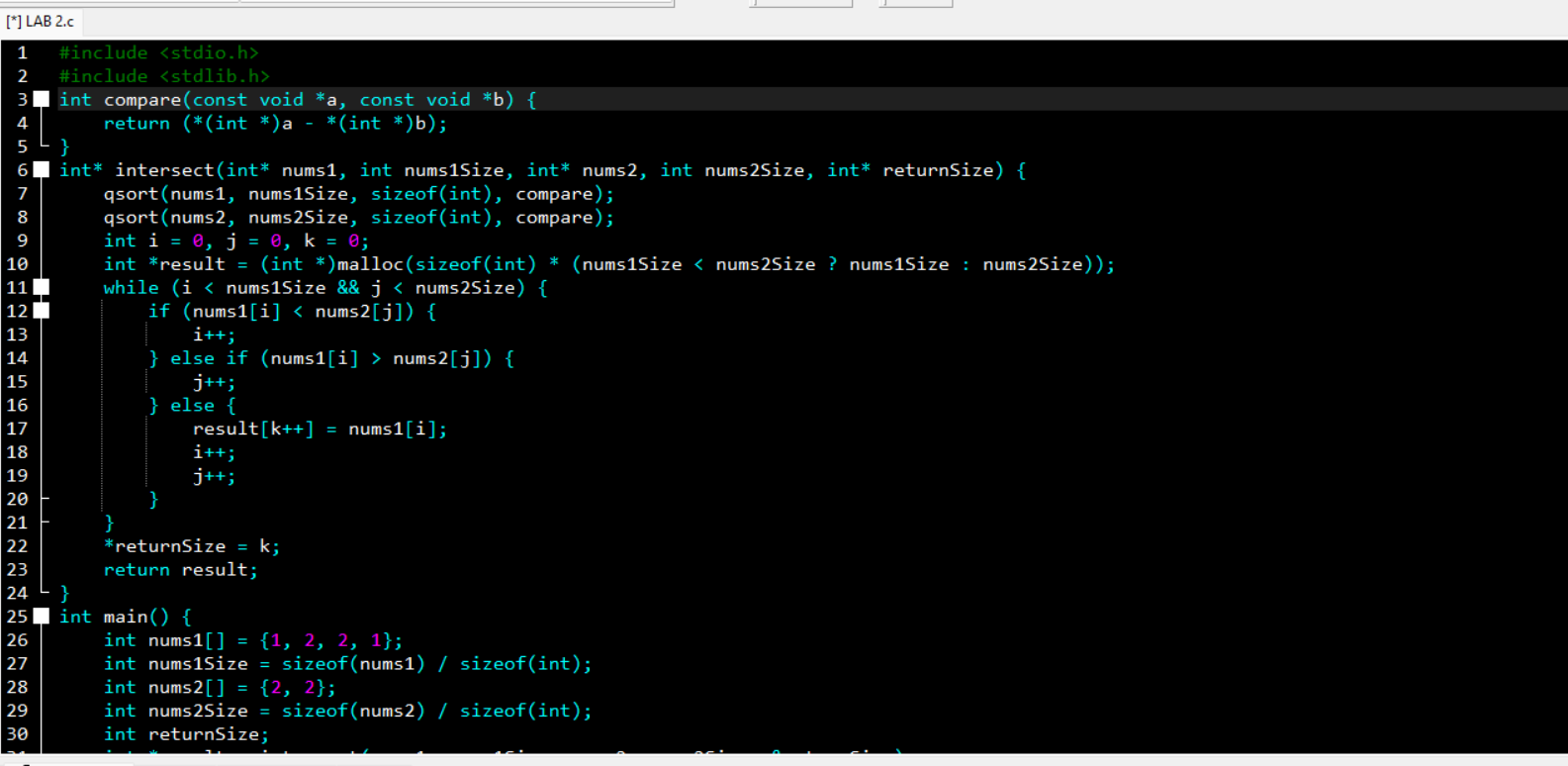


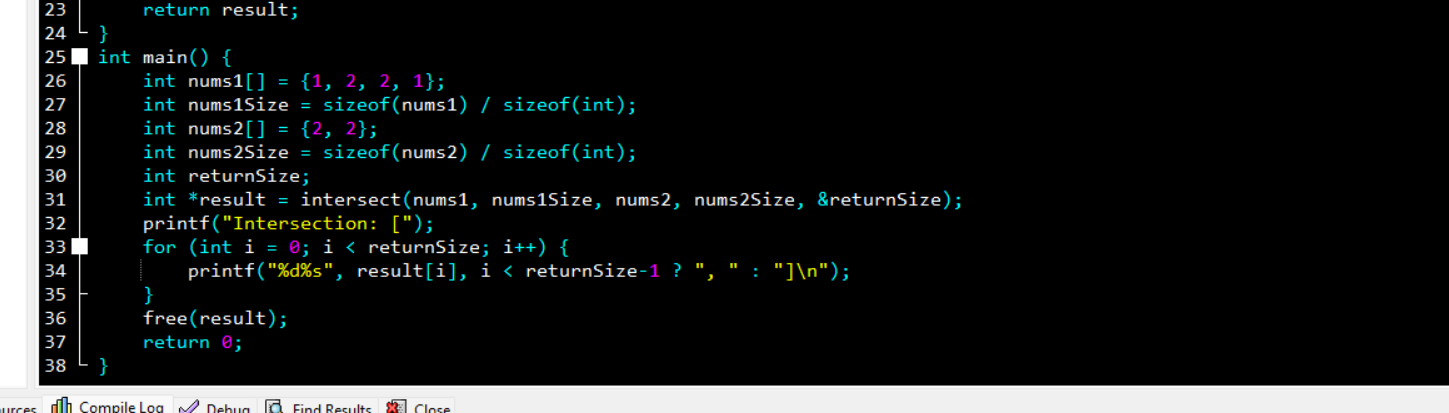
OUTPUT



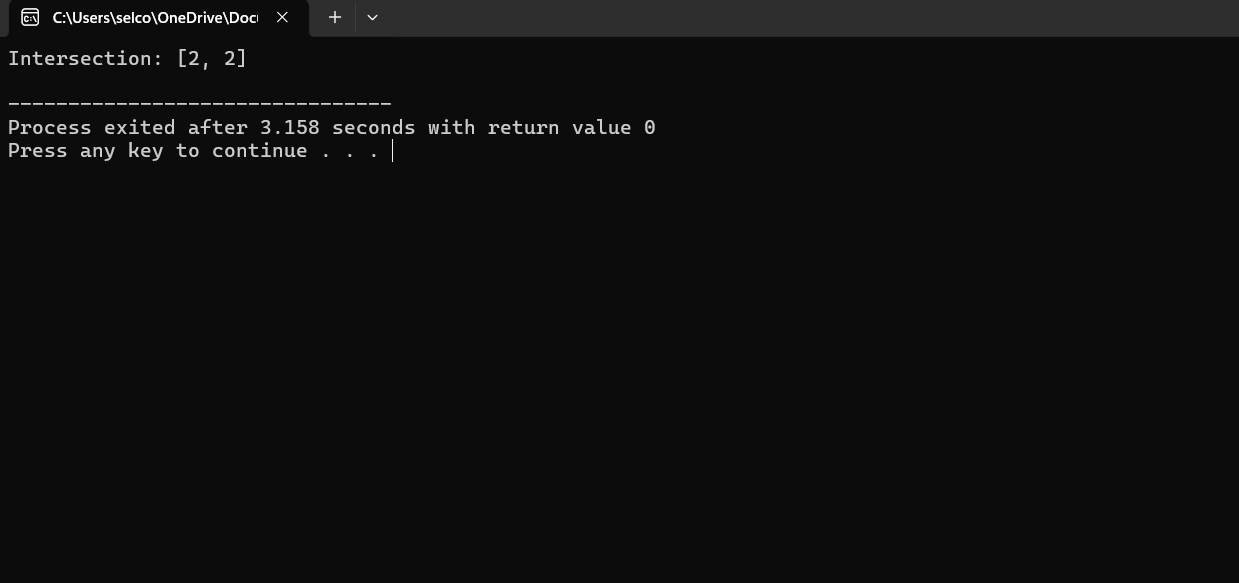
7)Given two integer arrays nums1 and nums2, return an array of their intersection. Each element in the result must appear as many times as it shows in both arrays and you may return the result in any order.

CODE:



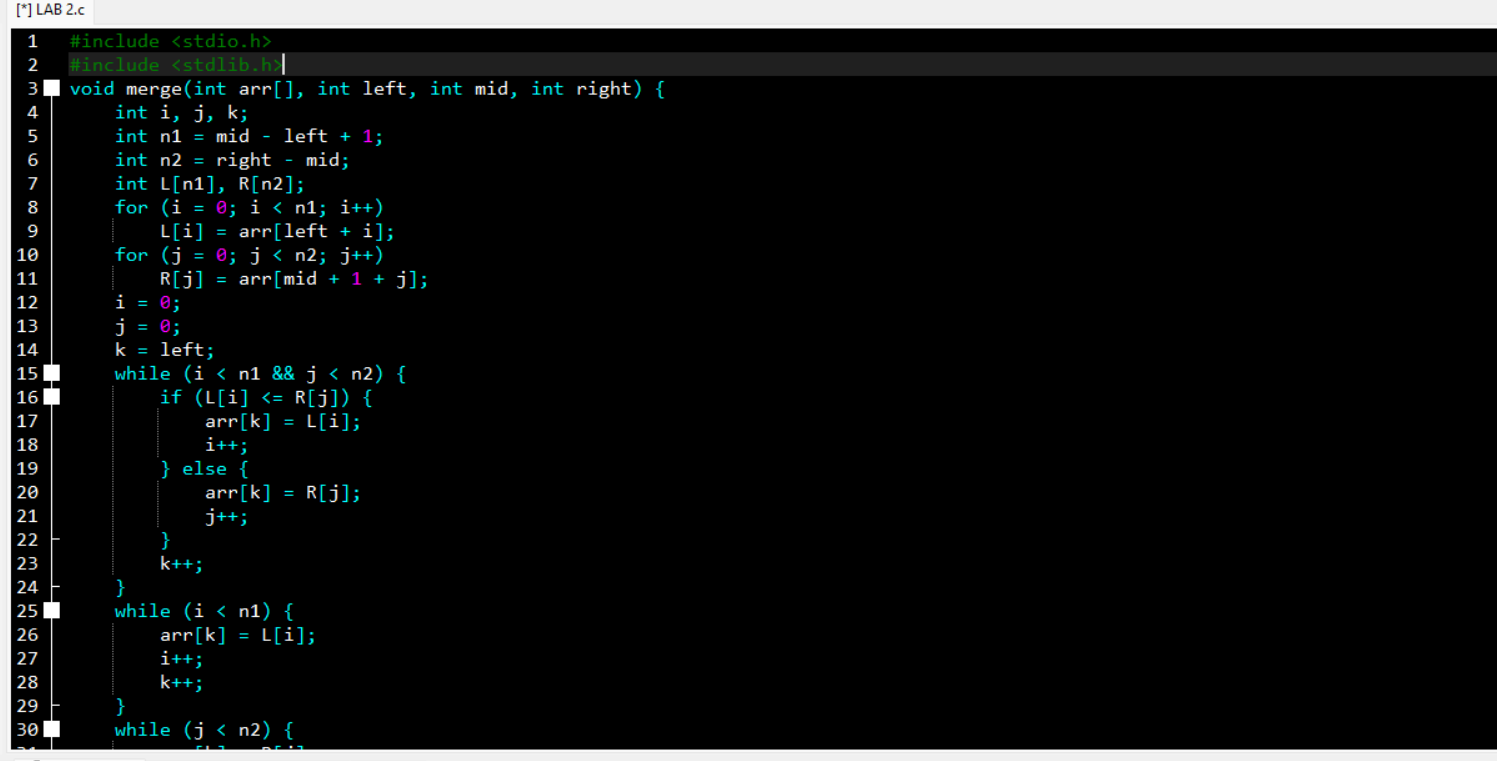


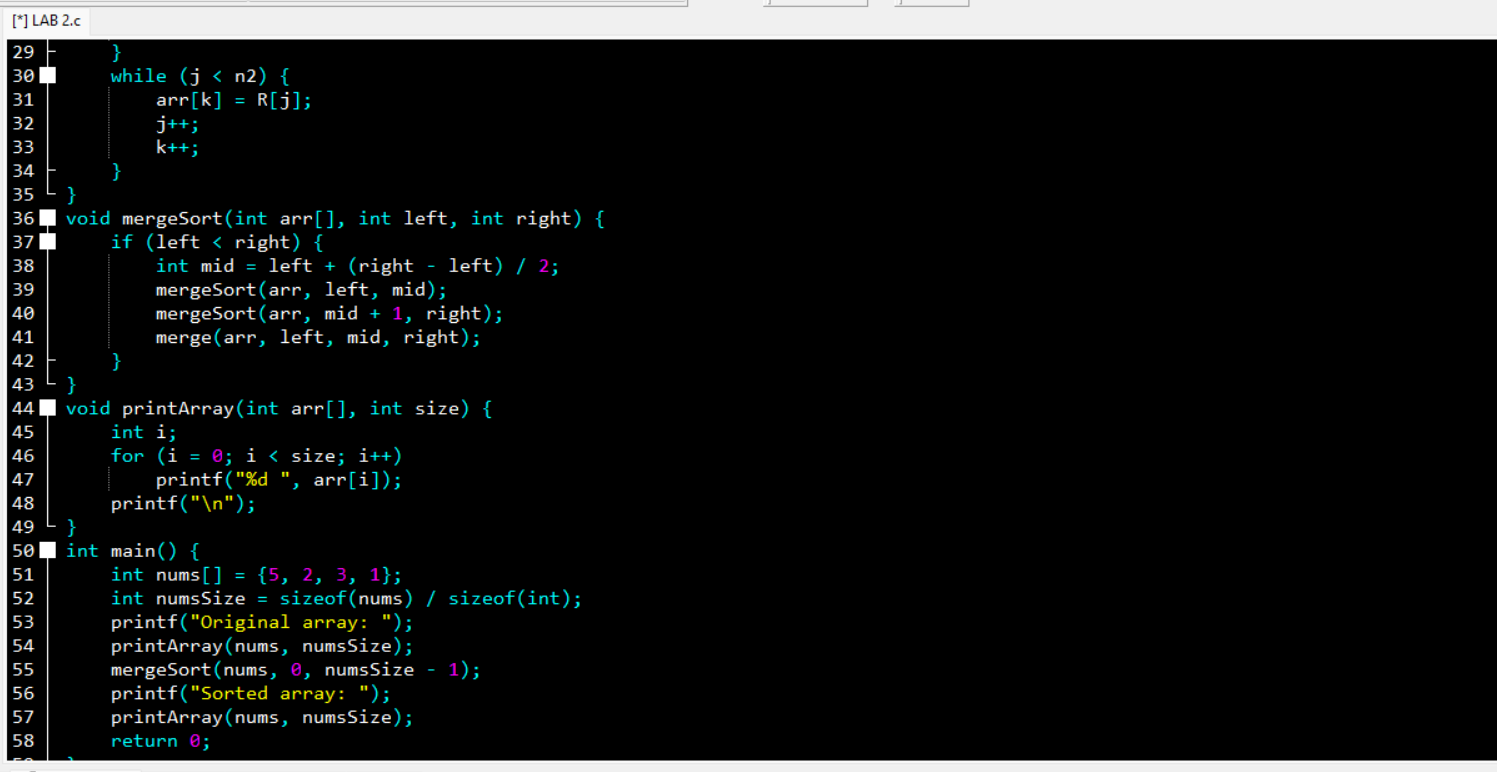
OUTPUT:

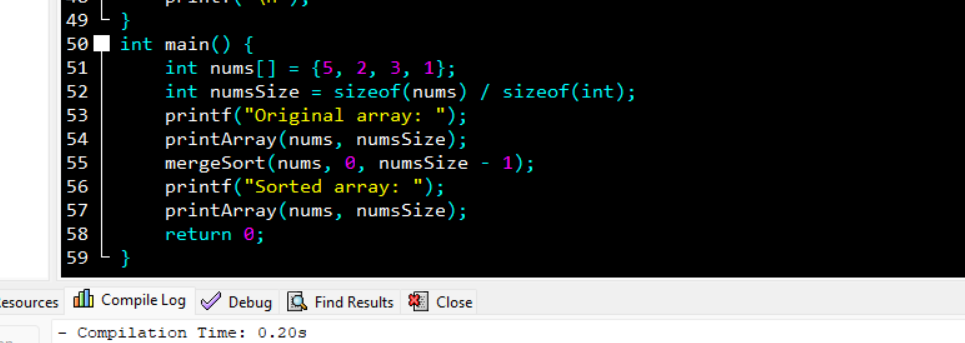


8)Given an array of integers nums, sort the array in ascending order and return it.You must solve the problem without using any built-in functions in O(nlog(n)) time complexity and with the smallest space complexity possible.

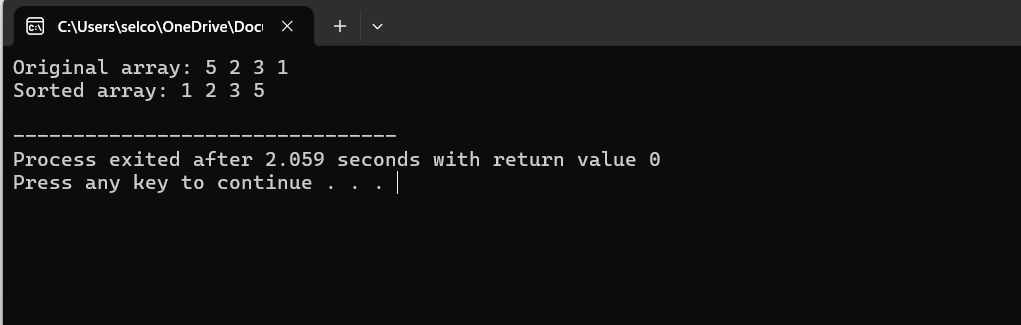
CODE:





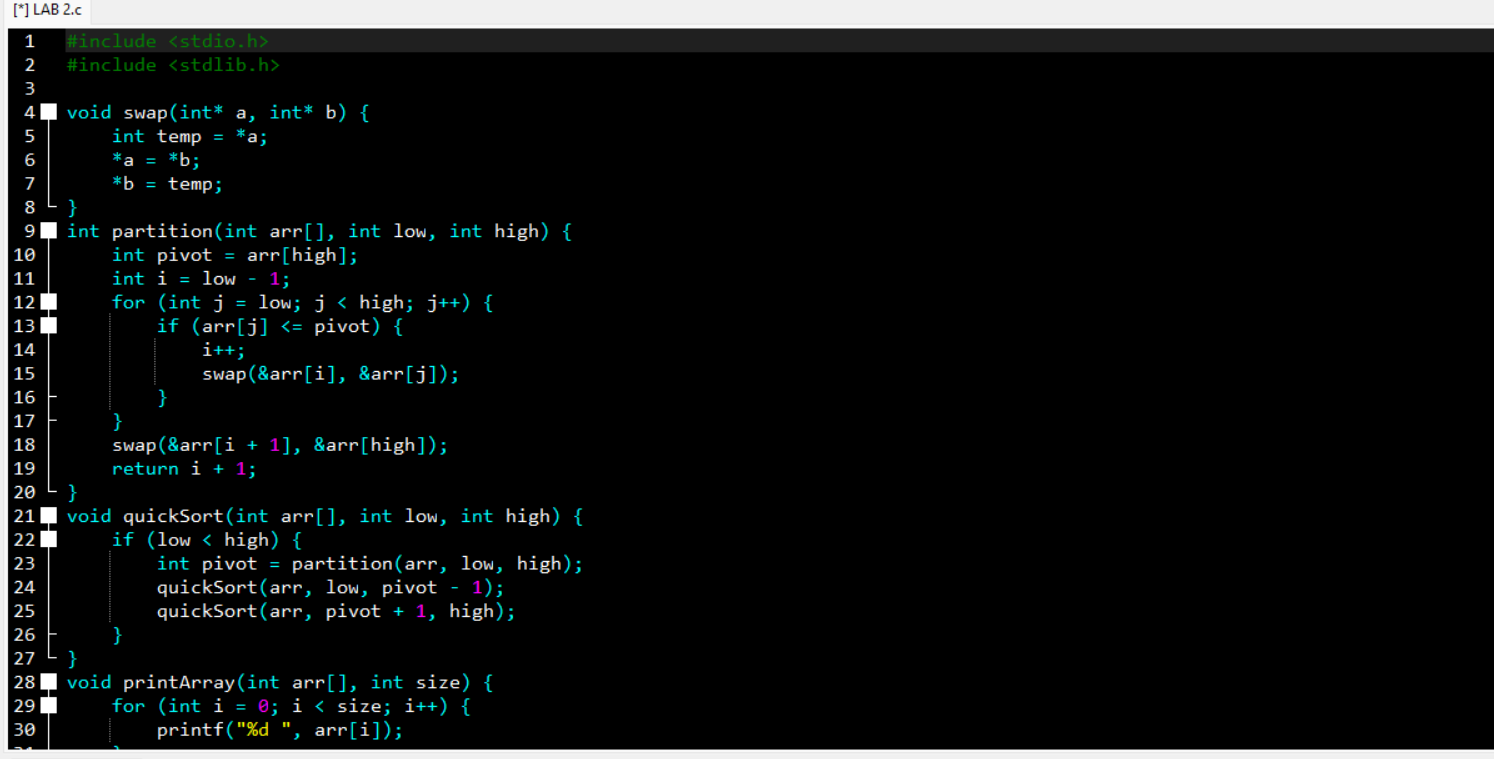


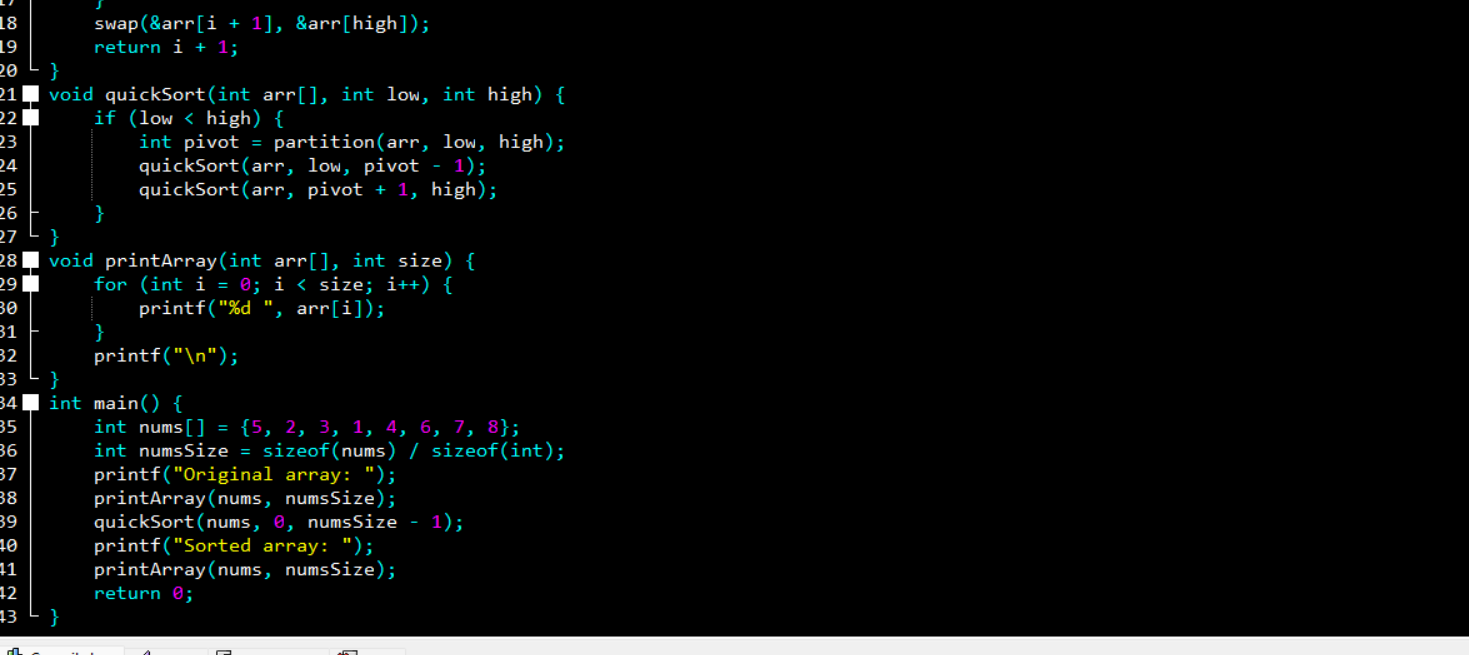
OUTPUT:



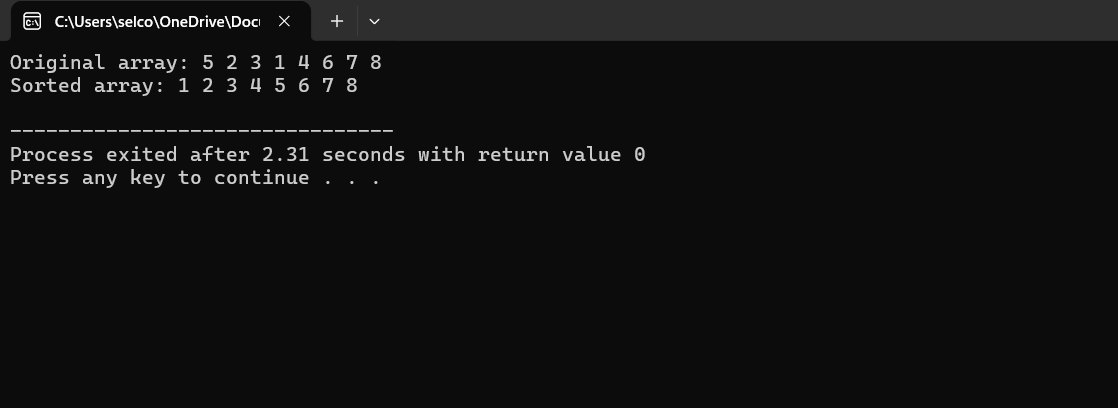
9)Given an array of integers nums, half of the integers in nums are odd, and the other half are even.

CODE;



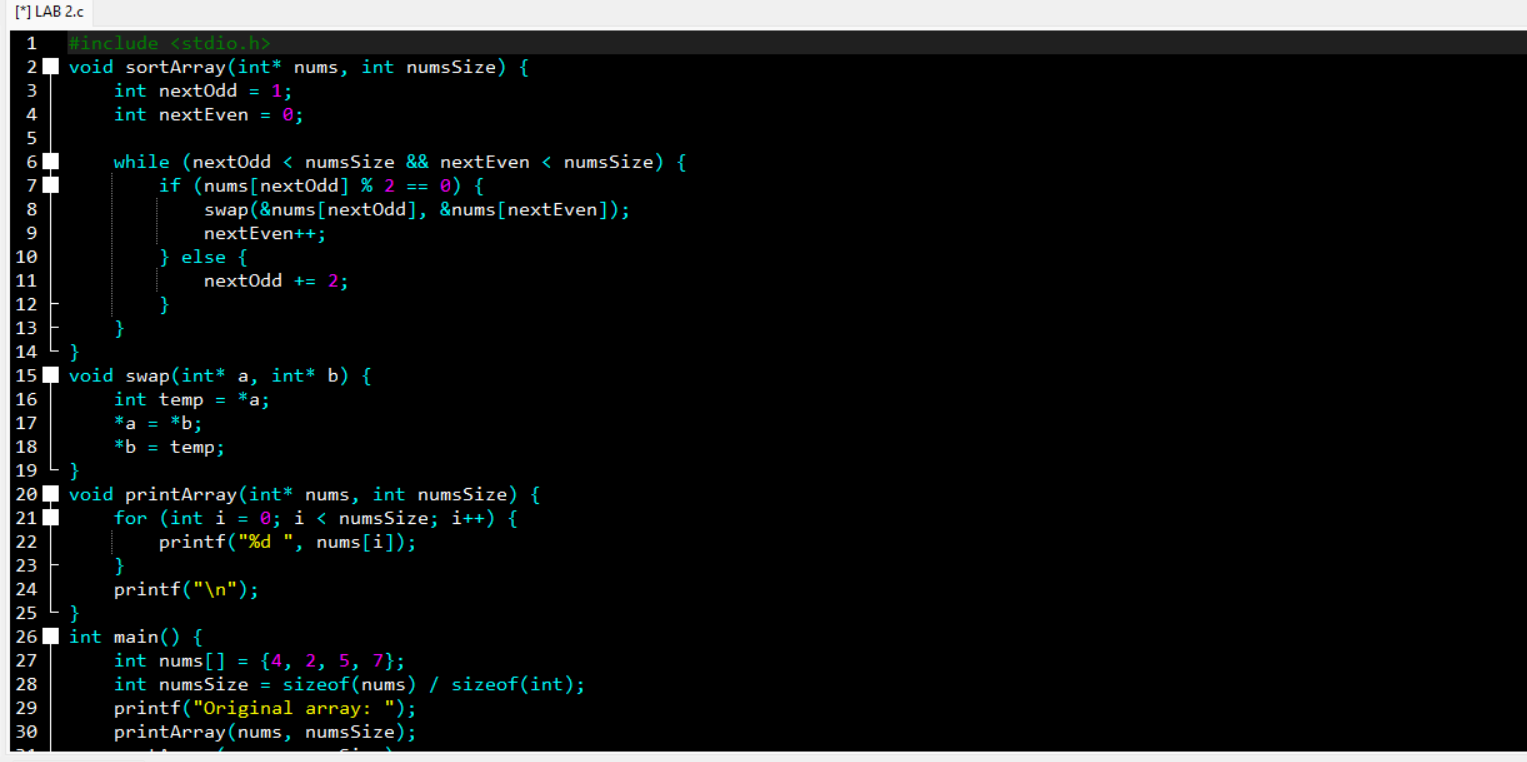


OUTPUT:



10)Sort the array so that whenever nums[i] is odd, i is odd, and whenever nums[i] is even, i is even. Return any answer array that satisfies this condition.

CODE:





OUTPUT:

