**VARUN BHARGAVA – 241010282**

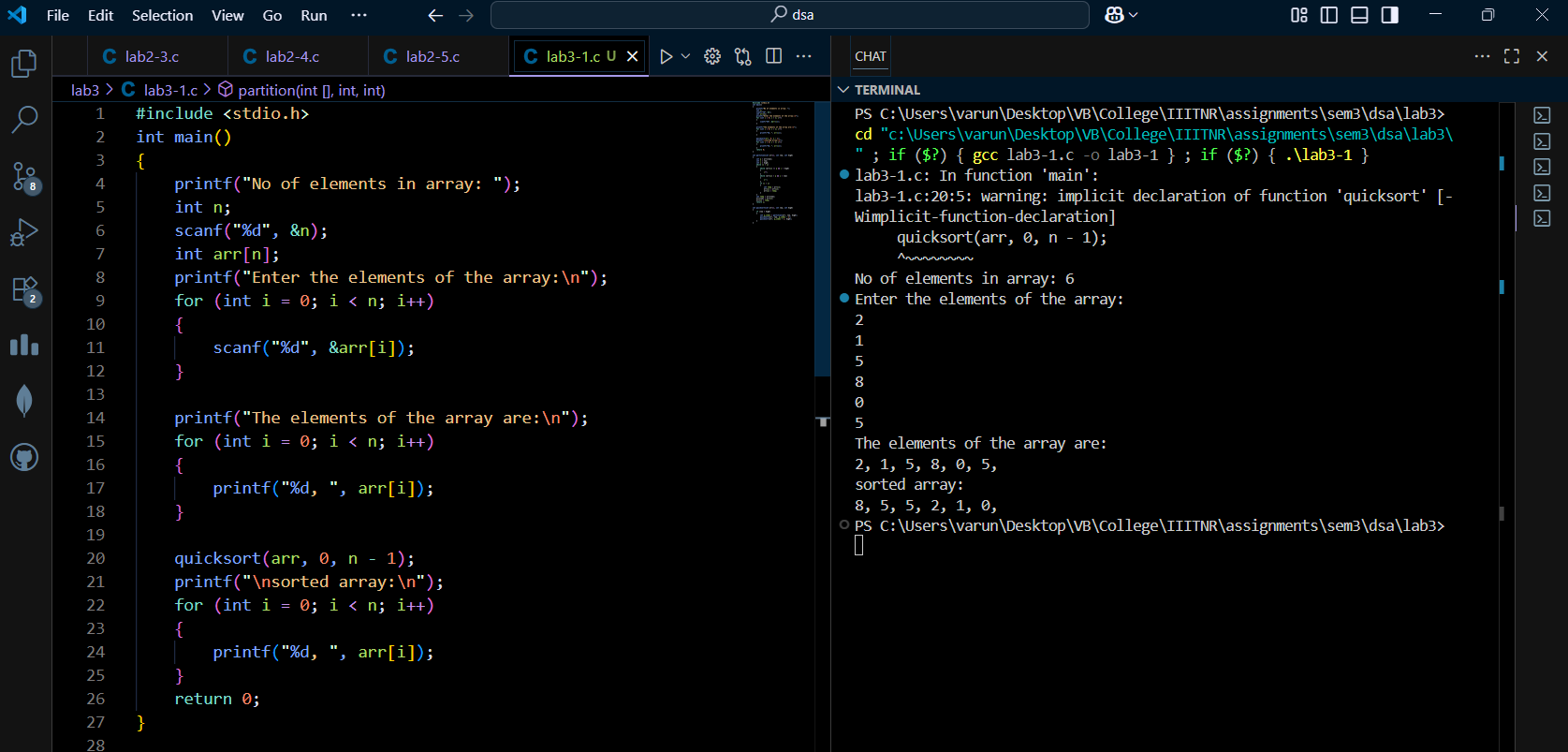
**DATA STRUCTURES TASK-3**

**Task 01: Quick Sort:**

( https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab3/lab3-1.c )

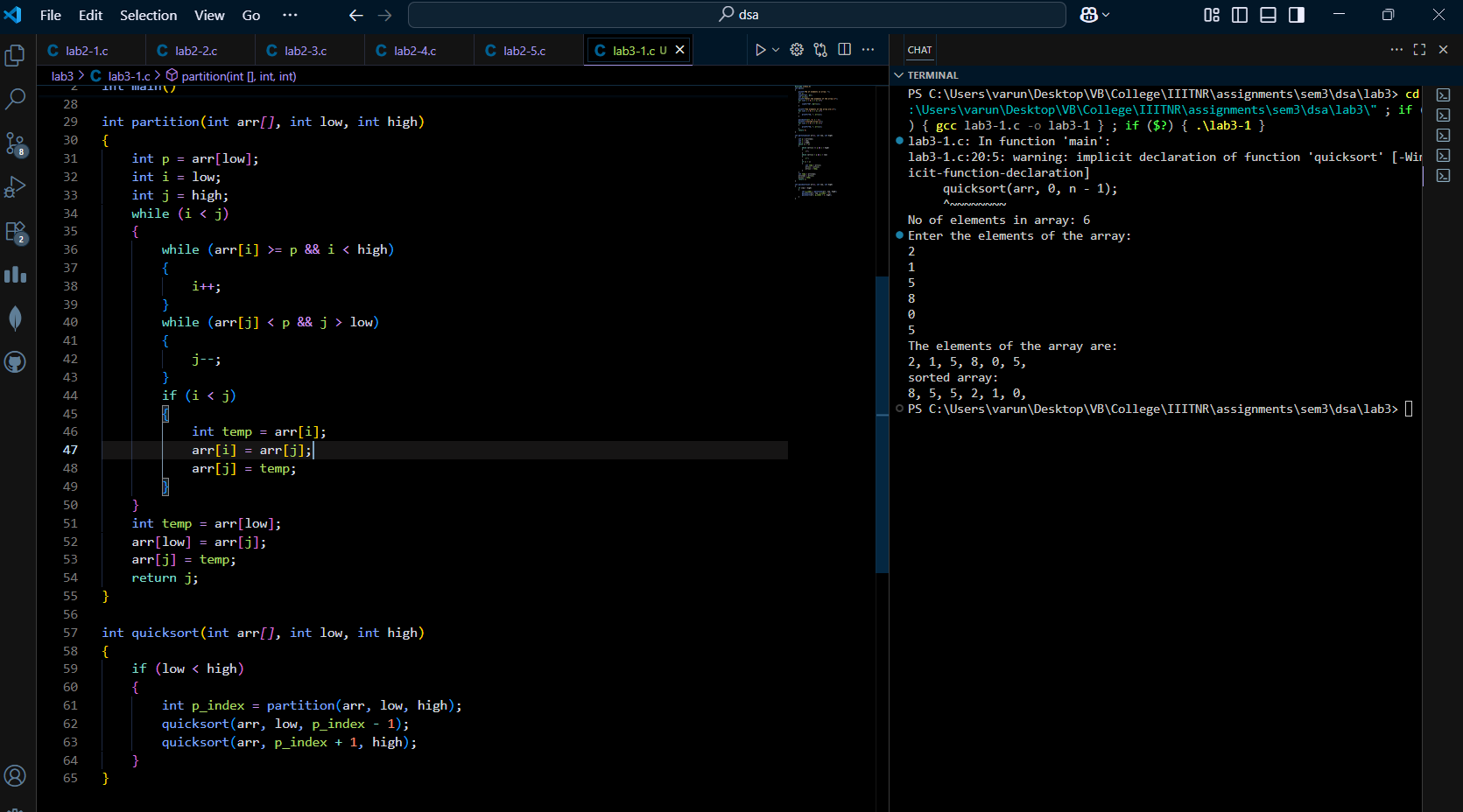
Write a program to perform the following operations using the Quick Sort algorithm:

1. Take user input to create an array of integers.



2. Sort the array in descending order using Quick Sort.

3. Display the sorted array.

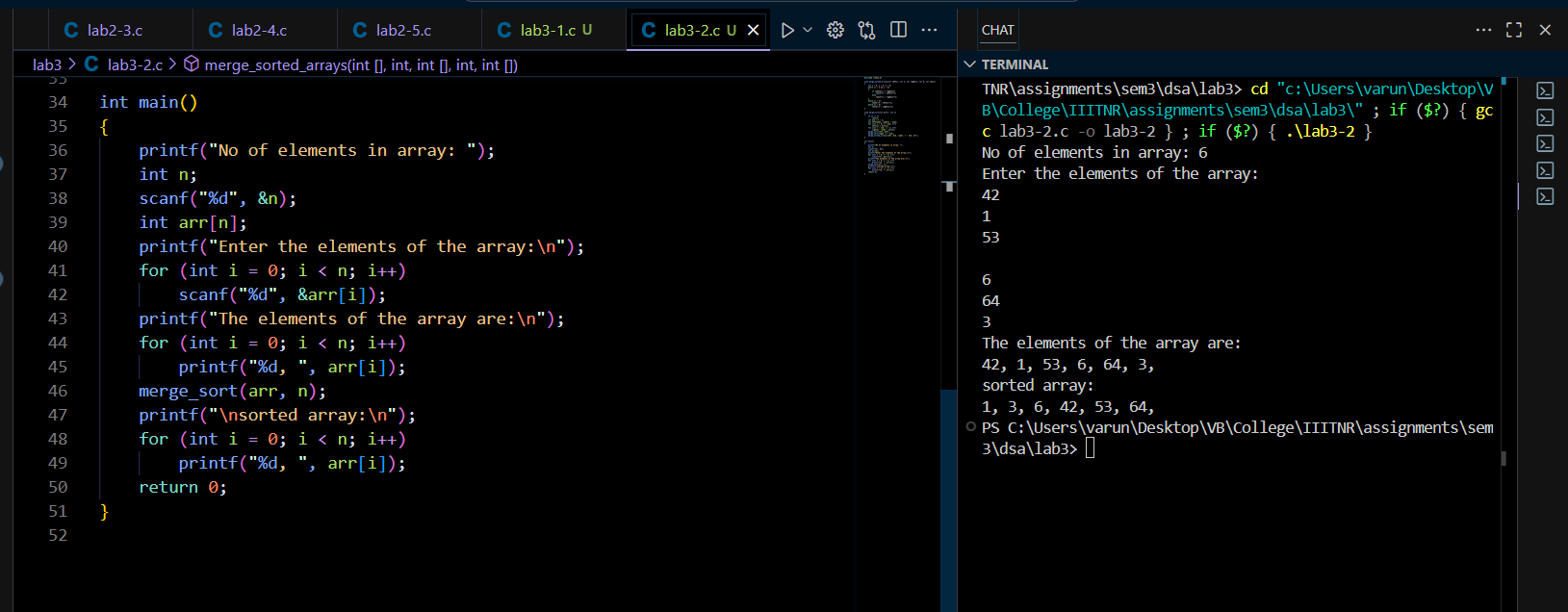


**Task 02: Merge Sort:**

( https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab3/lab3-2.c )

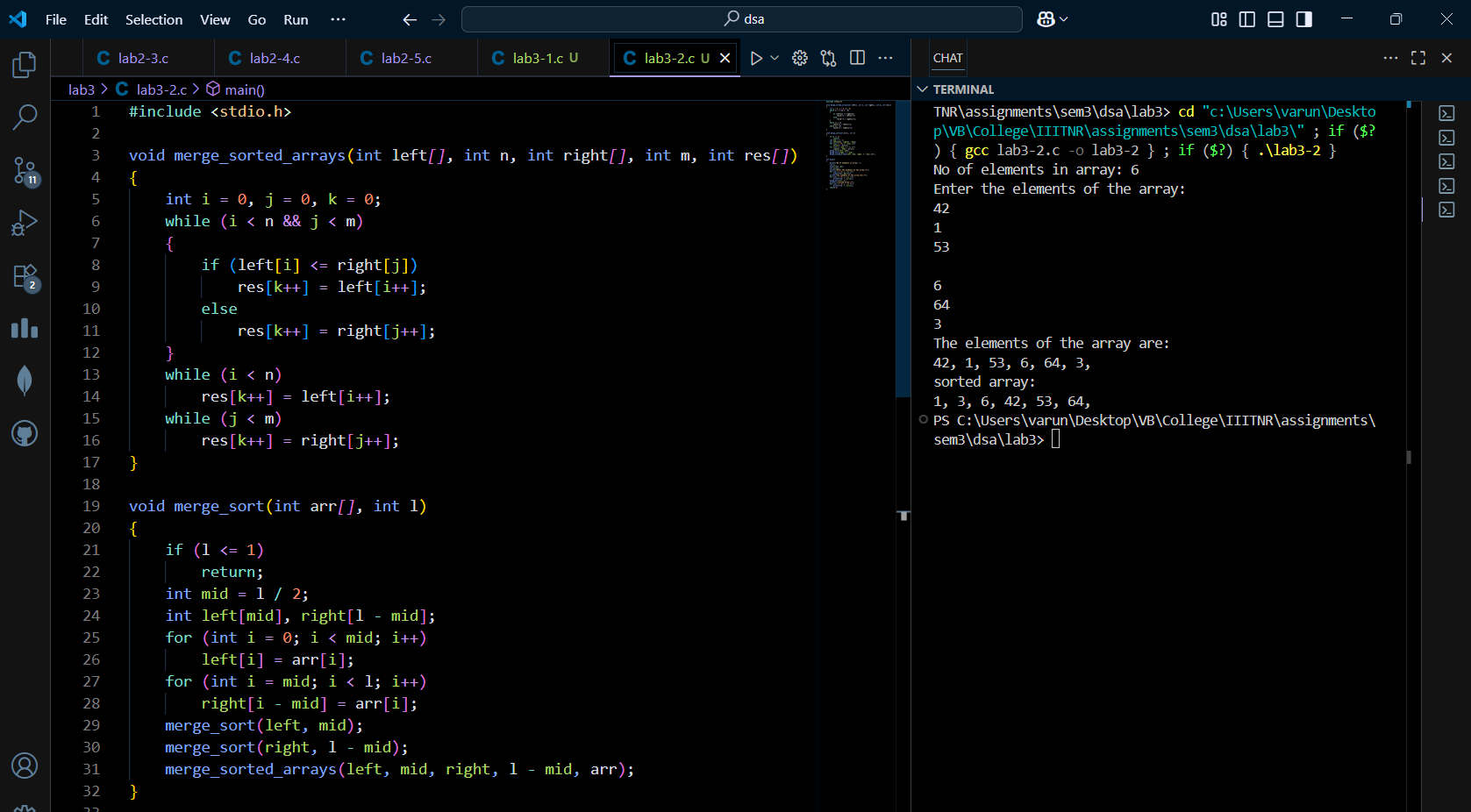
Write a program to perform the following operations using Merge Sort:

1. Take user input to create an array of integers.



2. Sort the array in ascending order using the Merge Sort algorithm.

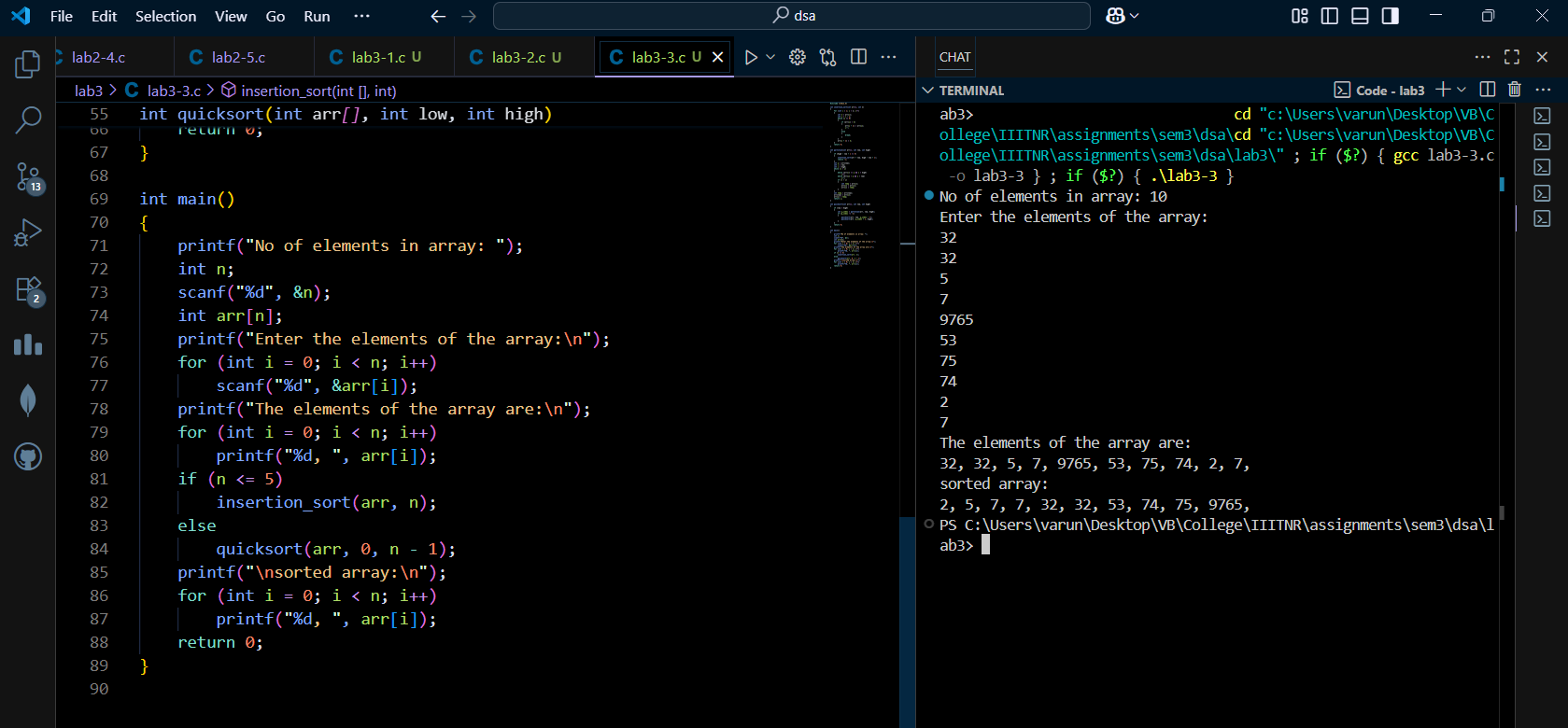
3. Display the sorted array.



**Task 03: Hybrid Sorting Algorithm:**

( https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab3/lab3-3.c )

Write a program to perform the following operations using a Hybrid Sorting Algorithm:

1. Take user input to create an array of integers.

2. Set a threshold value of 5 for switching from Quick Sort to Insertion Sort when

sorting small subarrays.

3. Implement Quick Sort to sort the array:

(a) If the size of the current subarray is greater than the threshold, continue with

Quick Sort.

(b) If the size of the current subarray is less than or equal to the threshold, use

Insertion Sort instead.

4. Sort the array in ascending order using this hybrid approach.

5. Display the sorted array after applying the hybrid sorting algorithm.

