ADVANCED C PROGRAMMING – MODULE 1 ASSESSMENT

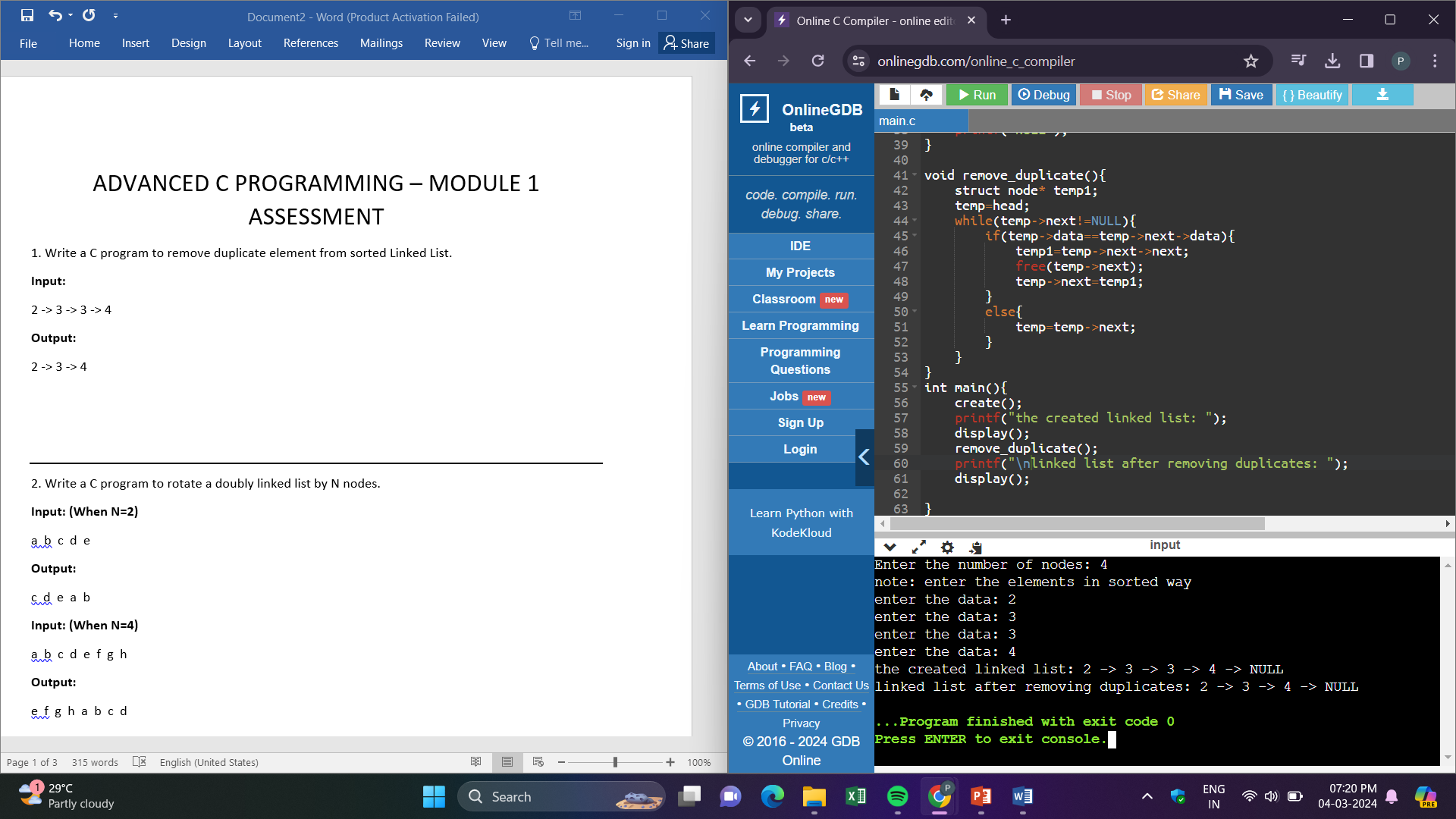
1. Write a C program to remove duplicate element from sorted Linked List.

**Input:**

2 -> 3 -> 3 -> 4

**Output:**

2 -> 3 -> 4



2. Write a C program to rotate a doubly linked list by N nodes.

**Input: (When N=2)**

a b c d e

**Output:**

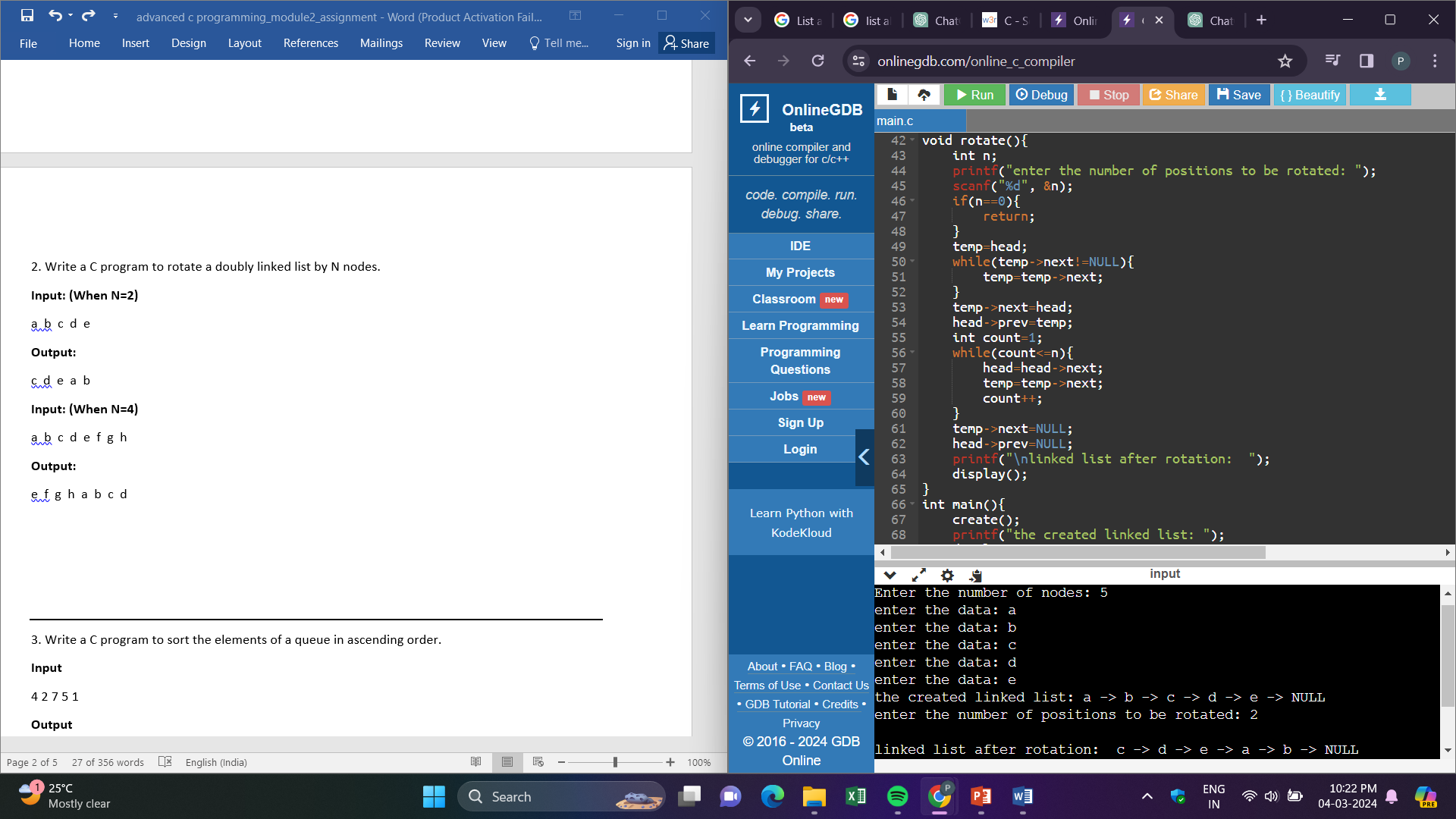
c d e a b

**Input: (When N=4)**

a b c d e f g h

**Output:**

e f g h a b c d



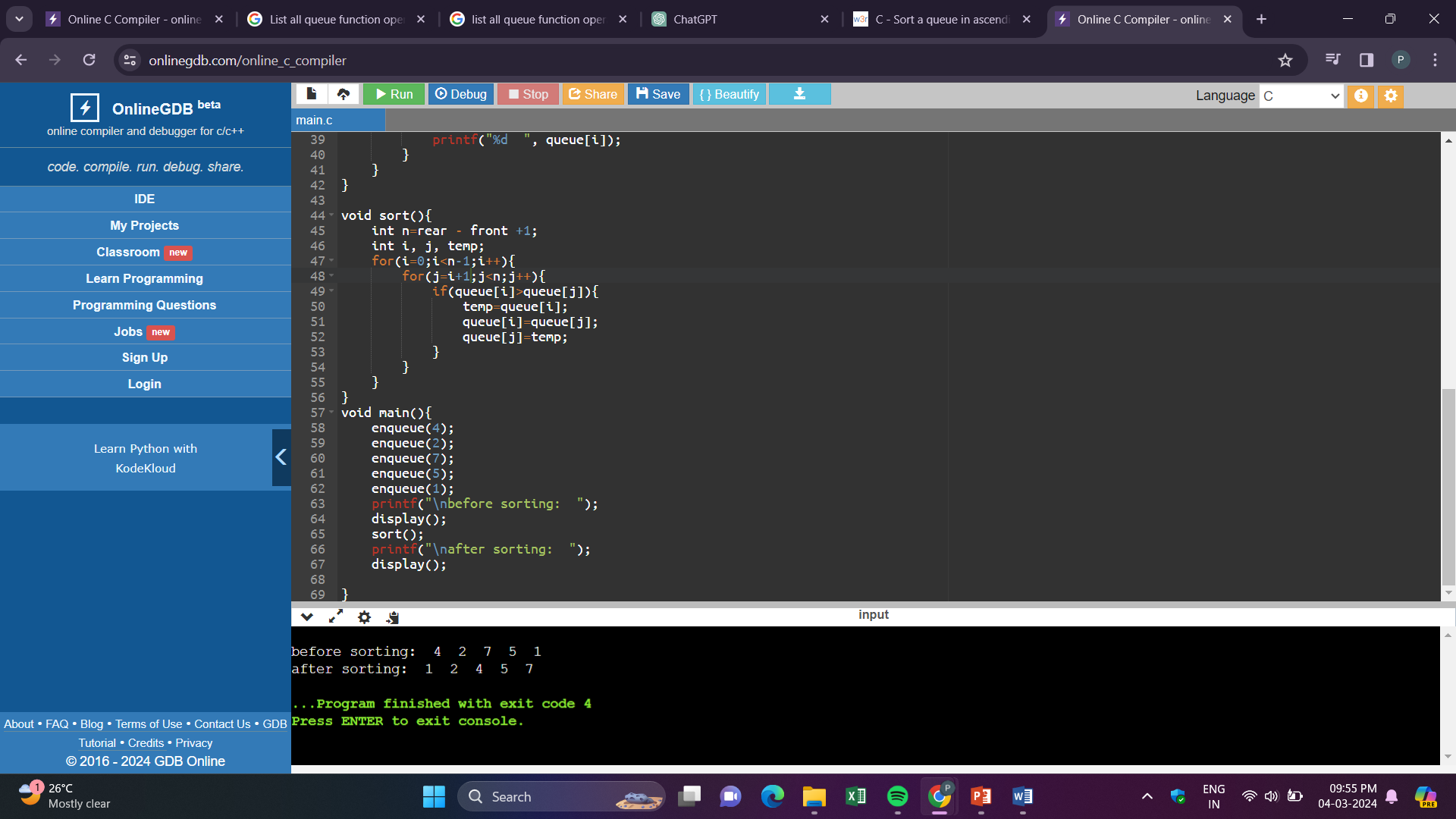
3. Write a C program to sort the elements of a queue in ascending order.

**Input**

4 2 7 5 1

**Output**

1 2 4 5 7



4. List all queue function operations available for manipulation of data elements in c

enqueue() – inserting elements in queue

dequeue() – removing the elements from queue

display() – to display all elements in a queue

peek() – gives the front node without dequeuing it

isfull() – checks if the queue if full

isempty() – checks if the queue is empty

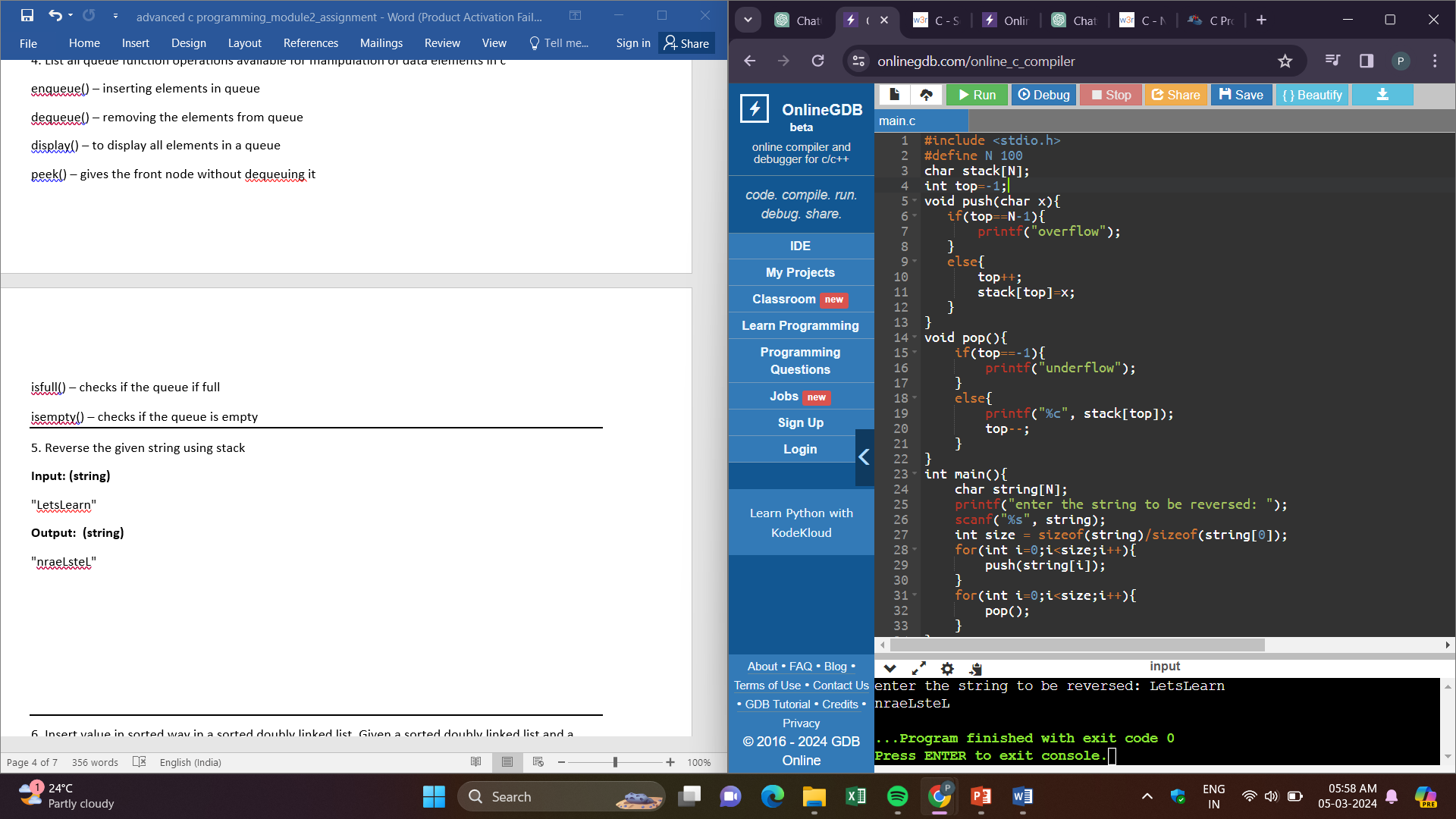
5. Reverse the given string using stack

**Input: (string)**

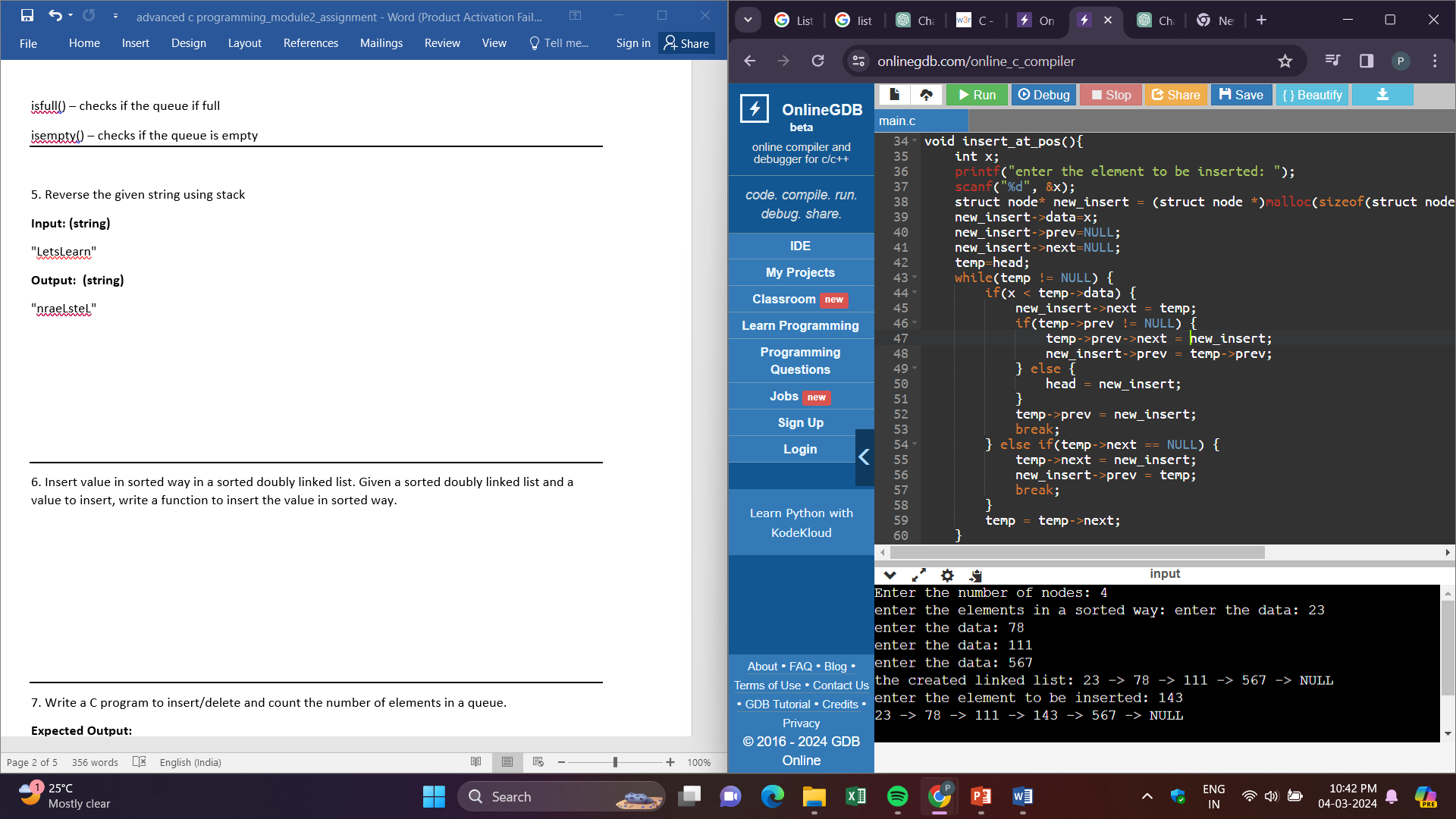
"LetsLearn"

**Output: (string)**

"nraeLsteL"



6. Insert value in sorted way in a sorted doubly linked list. Given a sorted doubly linked list and a value to insert, write a function to insert the value in sorted way.



7. Write a C program to insert/delete and count the number of elements in a queue.

**Expected Output:**

Initialize a queue!

Check the queue is empty or not? Yes

Number of elements in queue: 0

Insert some elements into the queue:

Queue elements are: 1 2 3

Number of elements in queue: 3

Delete two elements from the said queue:

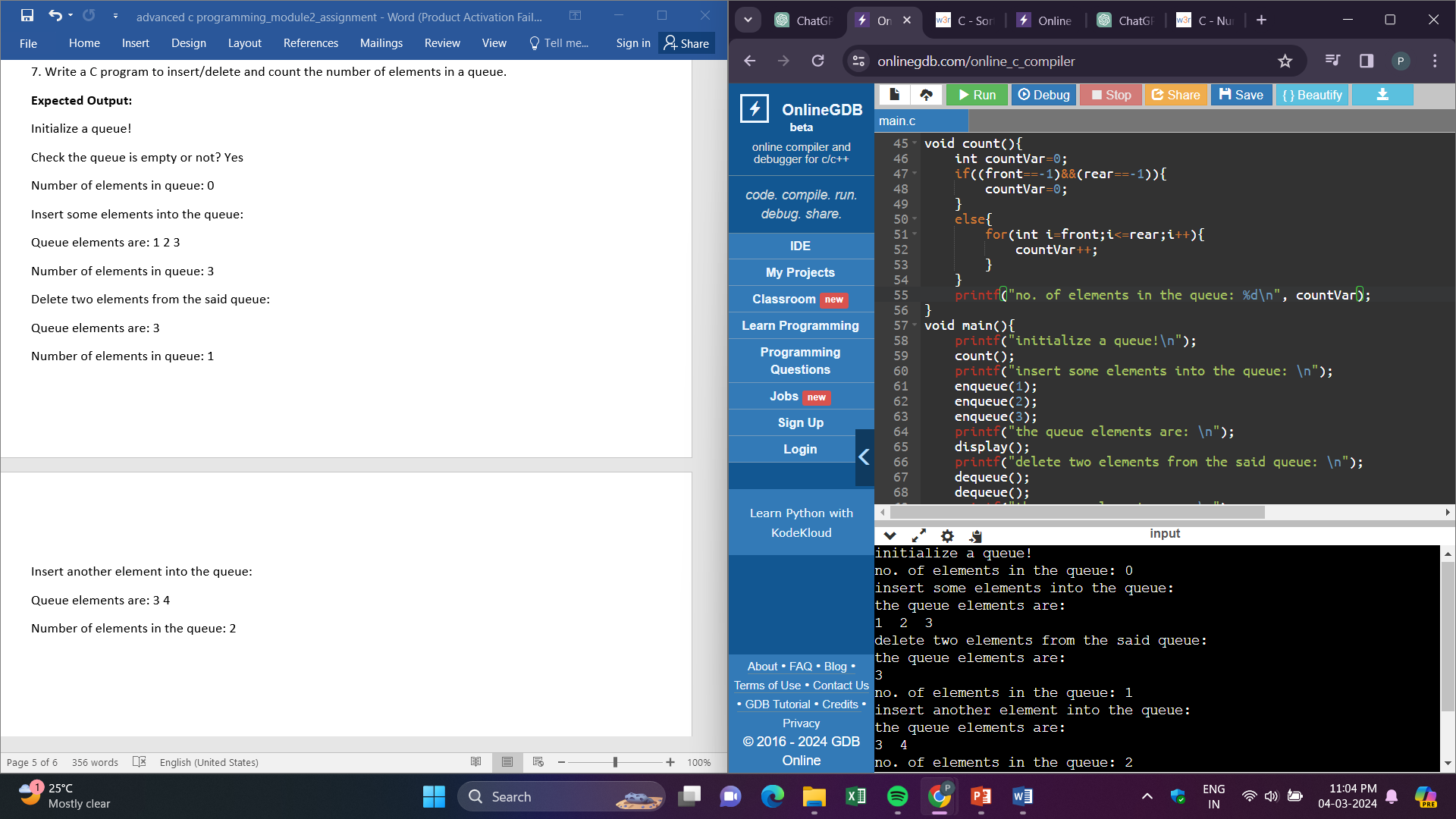
Queue elements are: 3

Number of elements in queue: 1

Insert another element into the queue:

Queue elements are: 3 4

Number of elements in the queue: 2



8. Write a C program to Find whether an array is a subset of another array.

**Input:**

arr1[] = {11, 1, 13, 21, 3, 7}, arr2[] = {11, 3, 7, 1}

**Output:**

arr2[] is a subset of arr1[]

**Input:**

arr1[] = {10, 5, 2, 23, 19}, arr2[] = {19, 5, 3}

**Output:**

arr2[] is not a subset of arr1[]

