|  |  |  |
| --- | --- | --- |
|  |  | **ISM 6225**  **Distributed Information systems** |

Assignment 1 – Programming Introduction

***Group Members:***

*Varaha Hari Hara Manoj Remela*

*Hemanthi Dabbirru*

*Angela Jones*

*Gerald Marks*

# Questions

## Question 1:

Professor Agrawal receives an array of integer *points* sorted in ascending order, the task is to find the initial and final index of a given *target* point’s value.

If the *target* point value is not found in the array of integers, return [-1,-1]

The professor had to leave for a conference at short notice and asked you to complete the task for him. He instructed you to limit the time complexity to O(n).

Example 1:

Input: marks = [5,6,6,9,9,12] target = 9

Output: [3,4]

Example 2:

Input: marks = [5,6,6,9,9,12] target = 10

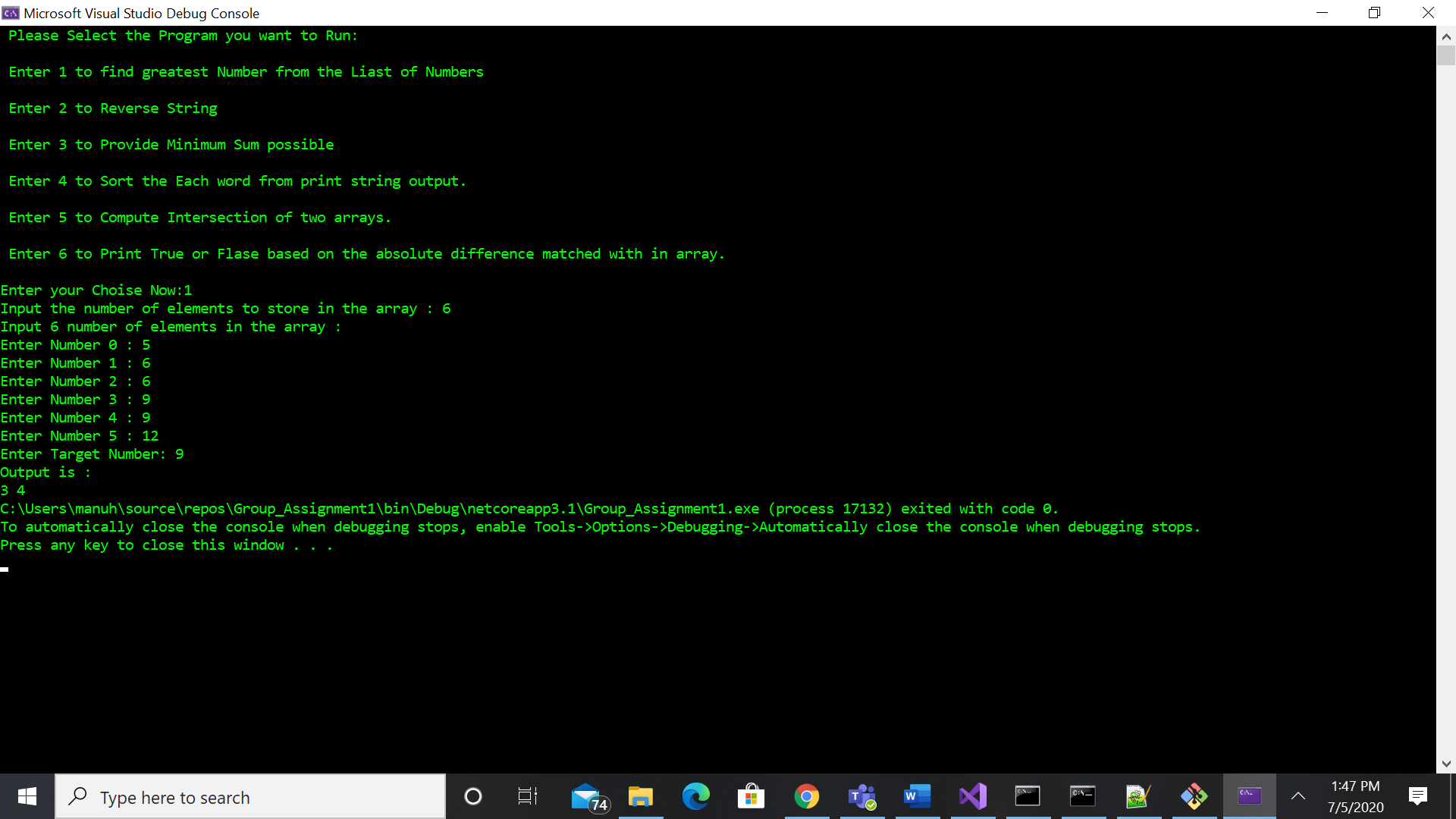
Output: [-1,-1]

**Note:** The algorithm’s runtime complexity must be O(n).

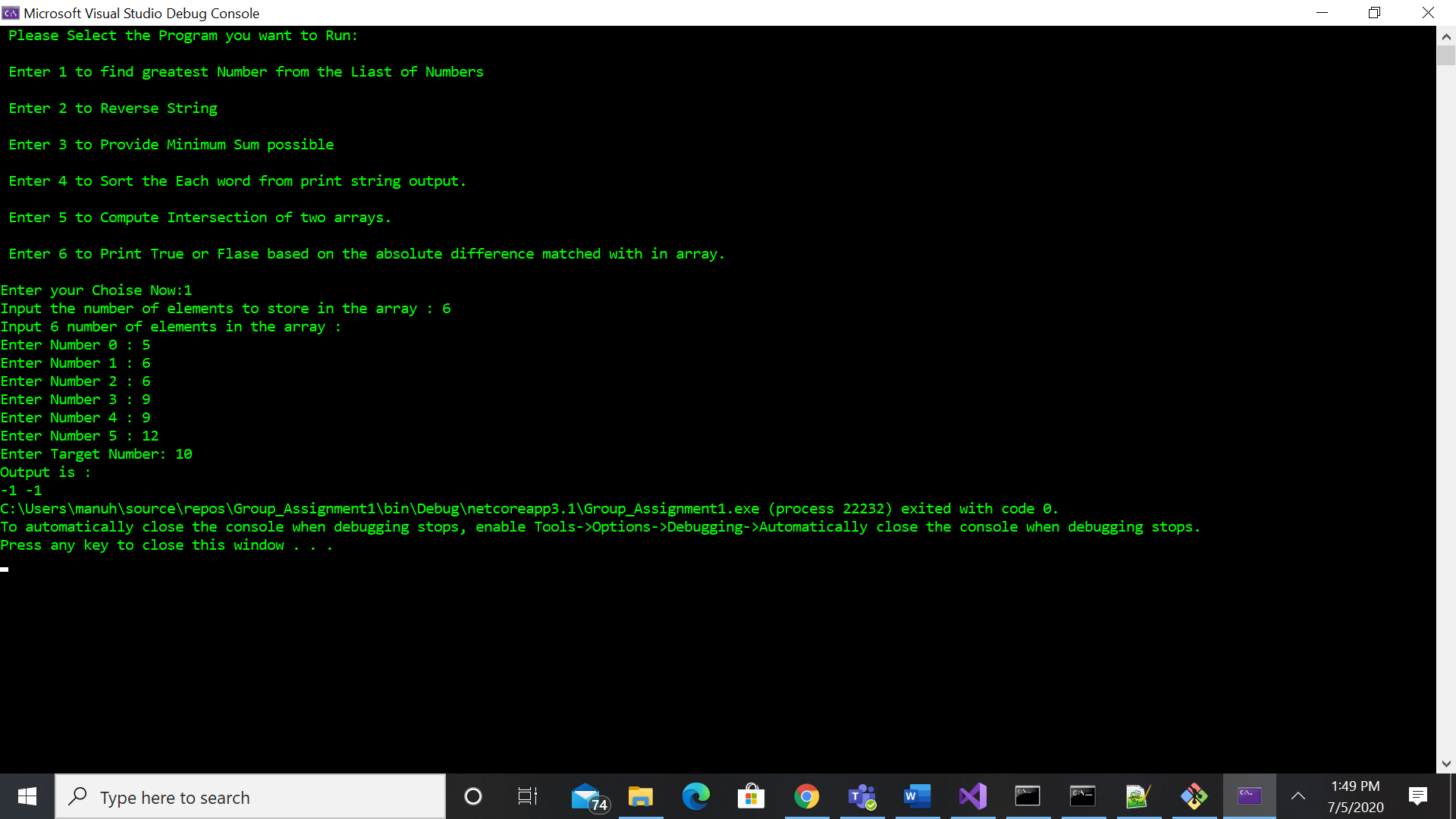
**public int[] targetRange(int[] marks, int target)**

**Program Output**: Please find the results for both the examples shared in above question.

Example-1:



Example-2:



## Question 2:

Rocky is not aware of the inbuilt functions to split and reverse a string. He is given a string and he needs to reverse the order of characters in each word within a sentence while still preserving whitespace and initial word order. He is not allowed to use any predefined split and reverse function. He is requesting you to complete the method for him.

Example:

Input : “University of South Florida”

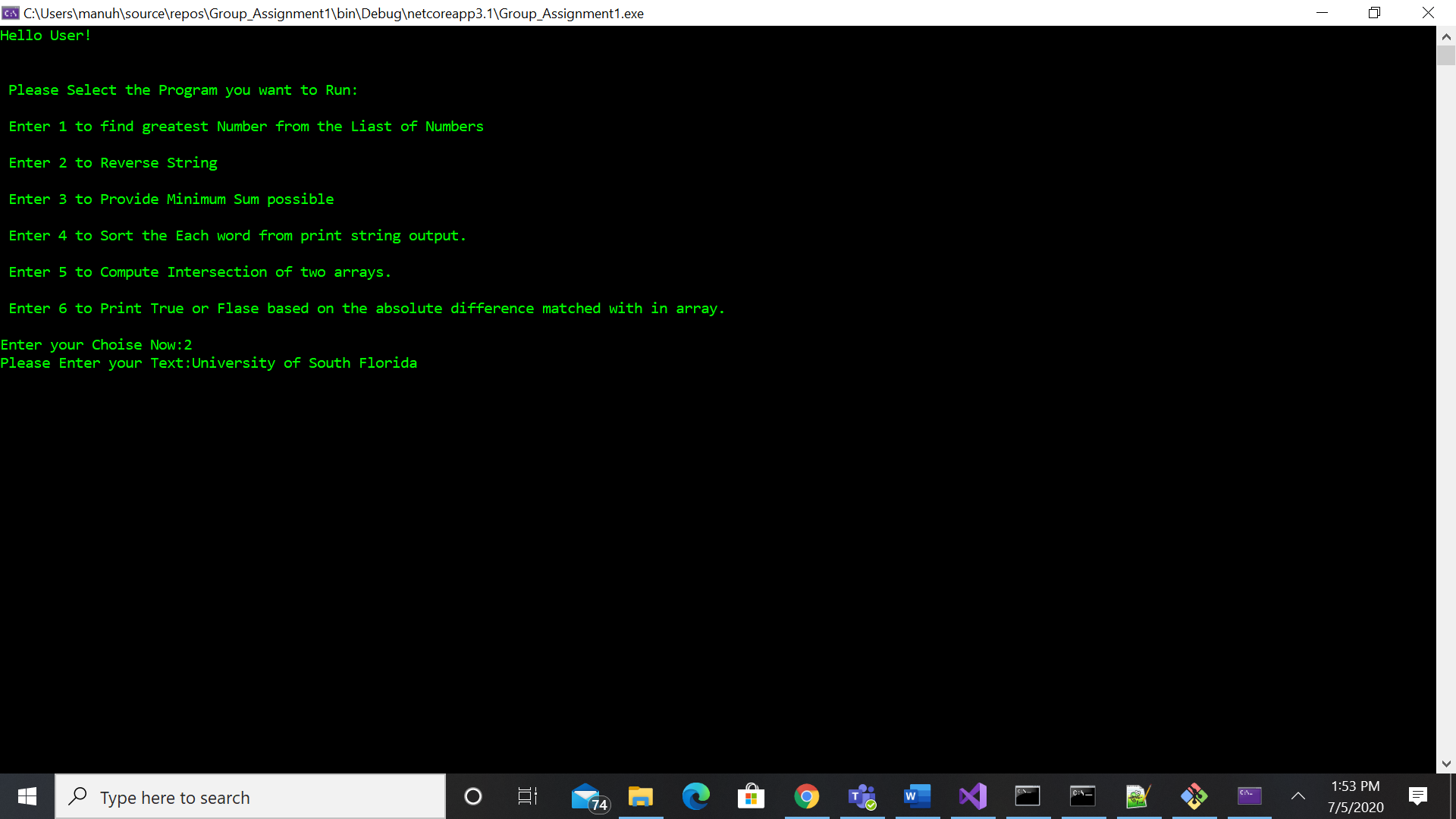
Output: “stisverinU fo htuoS adirolF”

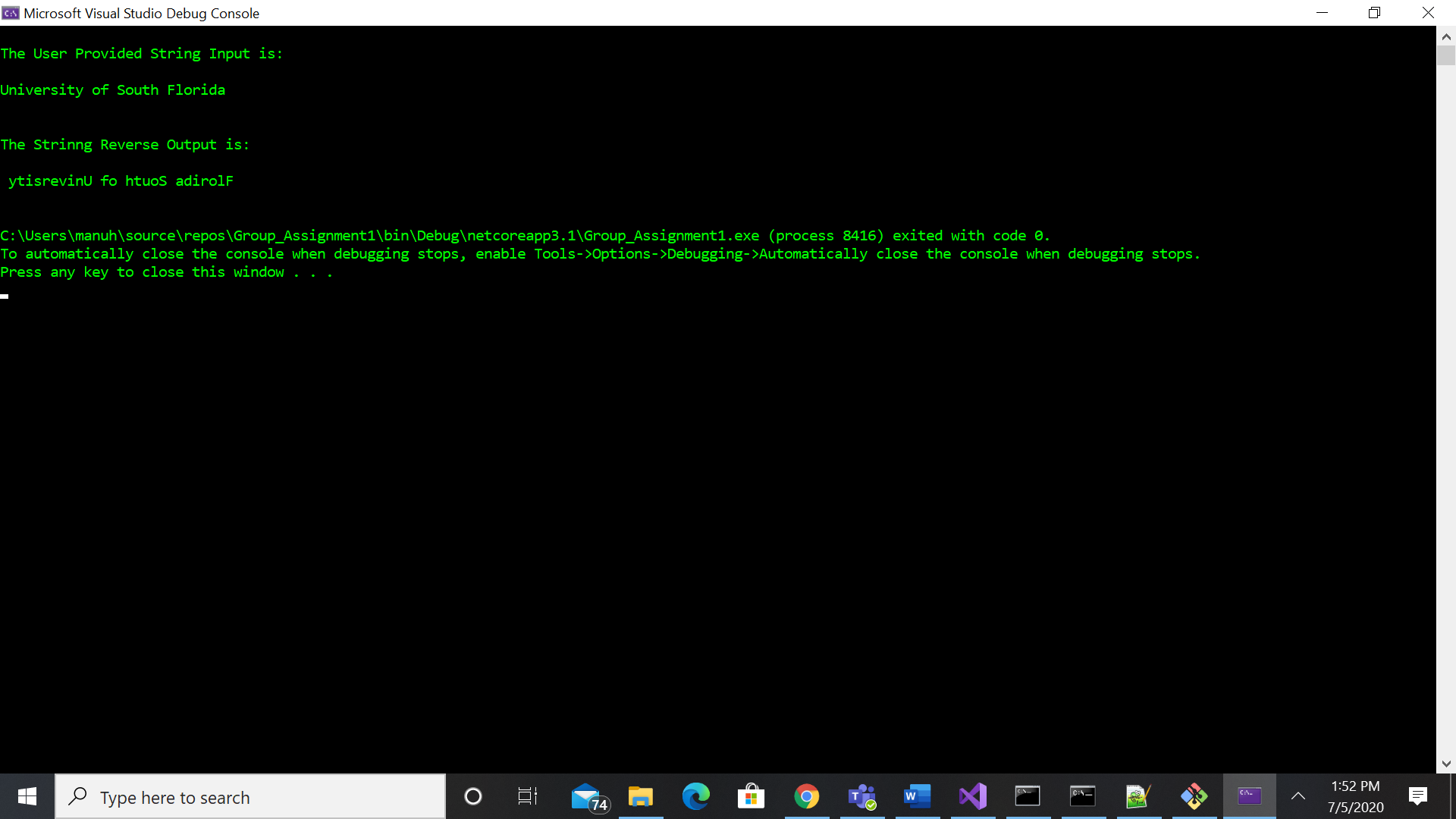
Note: Time complexity should be O(n) where n is the length of string

Hint: Create your own split and reverse function.

**Public static string StringReverse(string s)**

**Program Output: Please find the program output for the Example shared above.**





## Question 3:

Professor Stablein is given a sorted integer array. He needs to make the array elements distinct by increasing each value as needed, while minimizing the array sum. Professor Stablein thought this was an interesting exercise that the students might enjoy completing. Your job is to complete the method to print the minimum possible sum as output.

Example :

Input : arr[] = {2,2,3,5,6}; Output: 20

Explanation: Array is transformed to {2,3,4,5,6}. Sum becomes 2+3+4+5+6=20

Input arr[]= {40,40}; Output: 81

Explanation: Array is transformed to {40,41}. Sum becomes 81

Input: arr[] = {4,5,6,9}; Output : 24

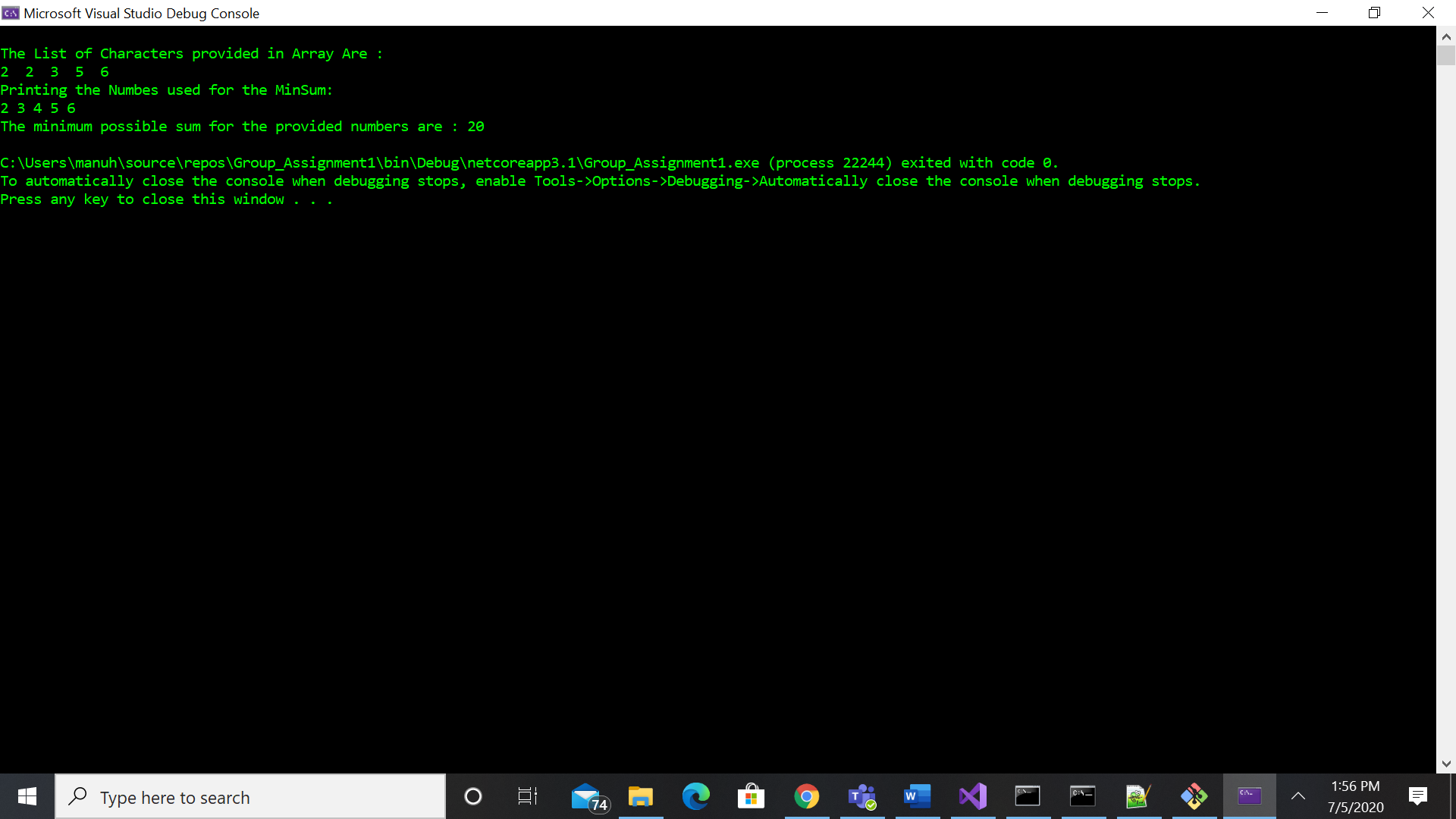
Explanation: As the array is sorted and has distinct elements. So result is sum of each elements in the array.

Time Complexity: O(n)

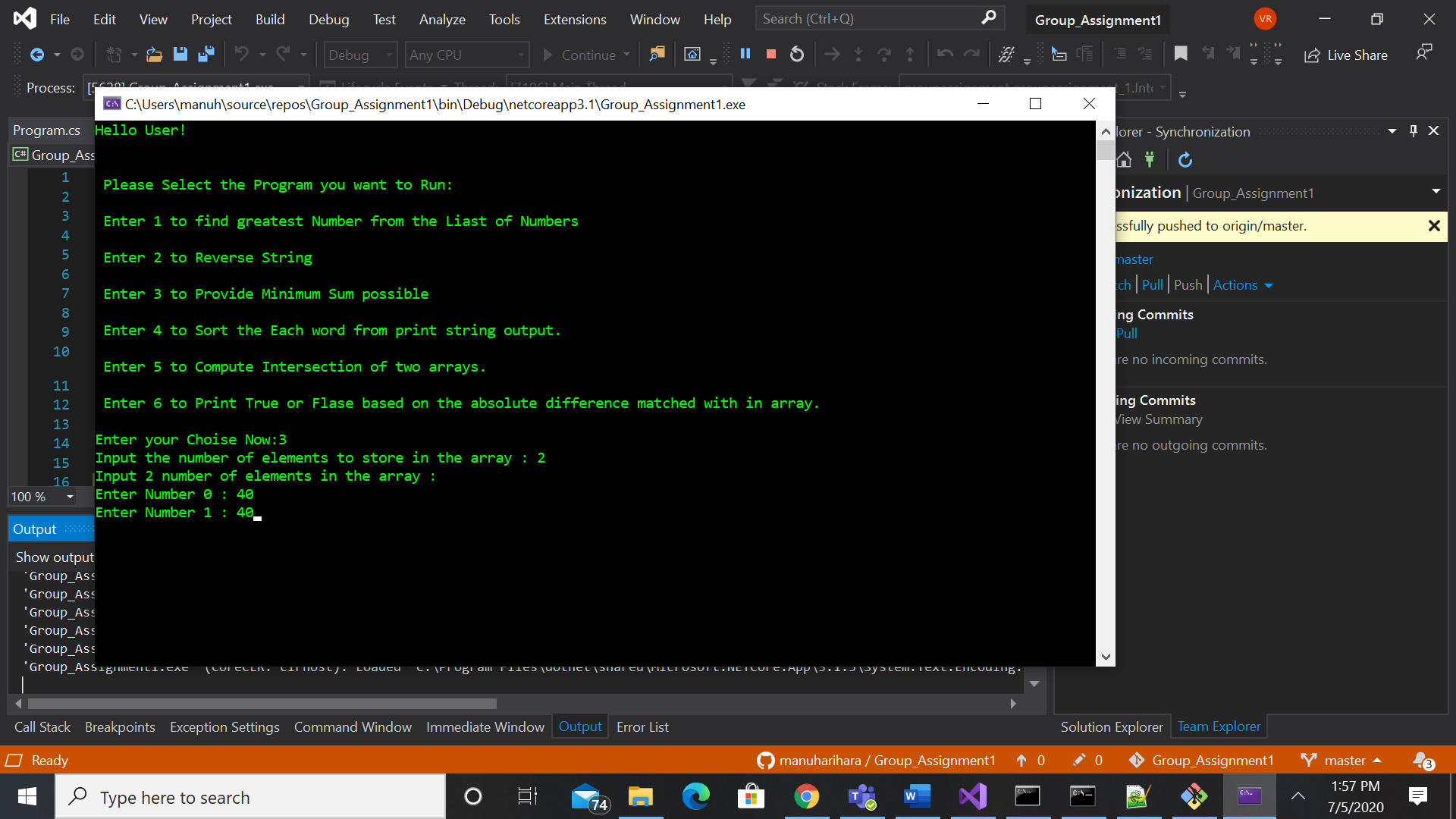
**public static int minSum(int[] arr)**

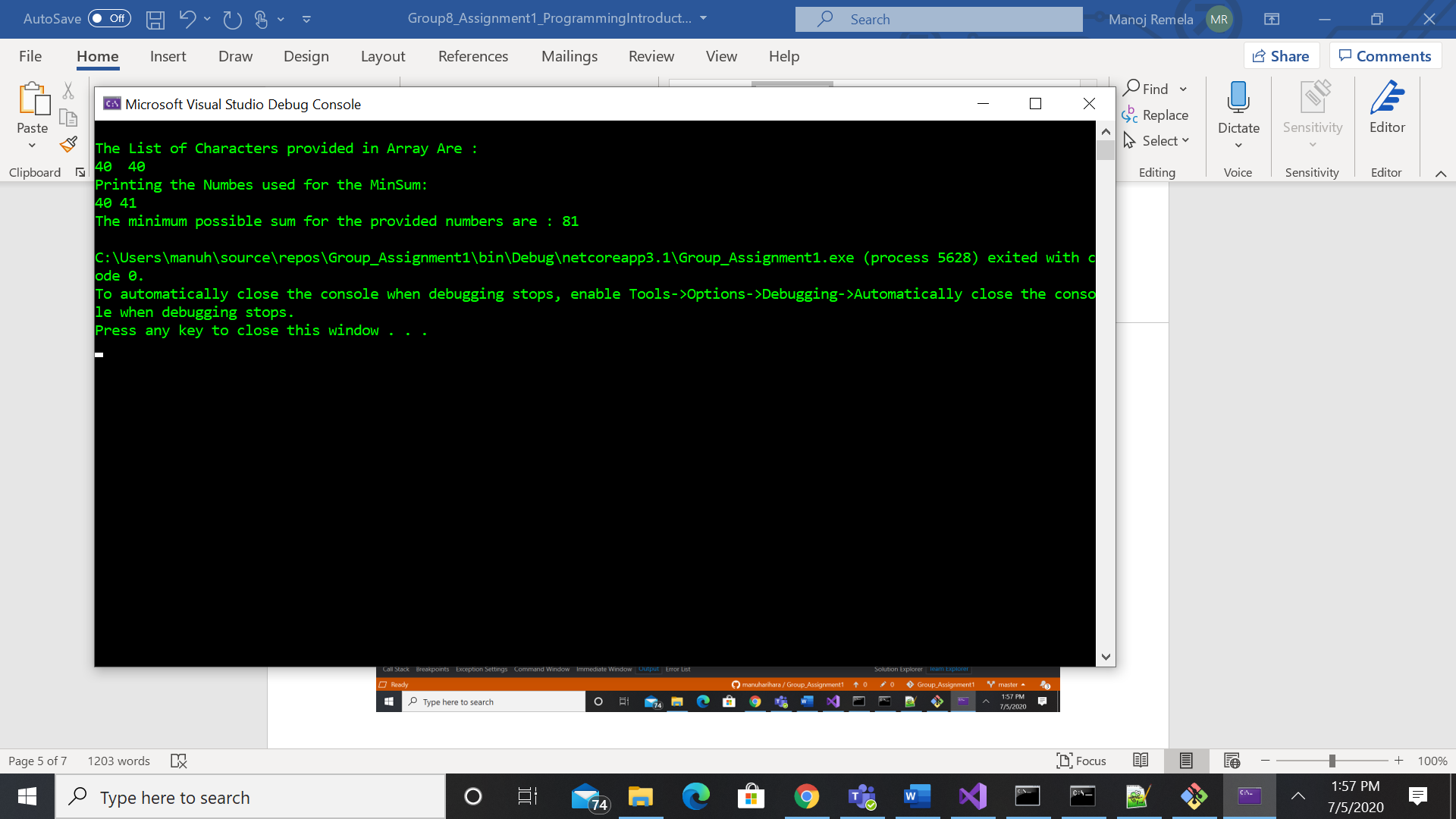
**Program Output: Please find the program output for all the Examples shared above.**

**Example-1:**

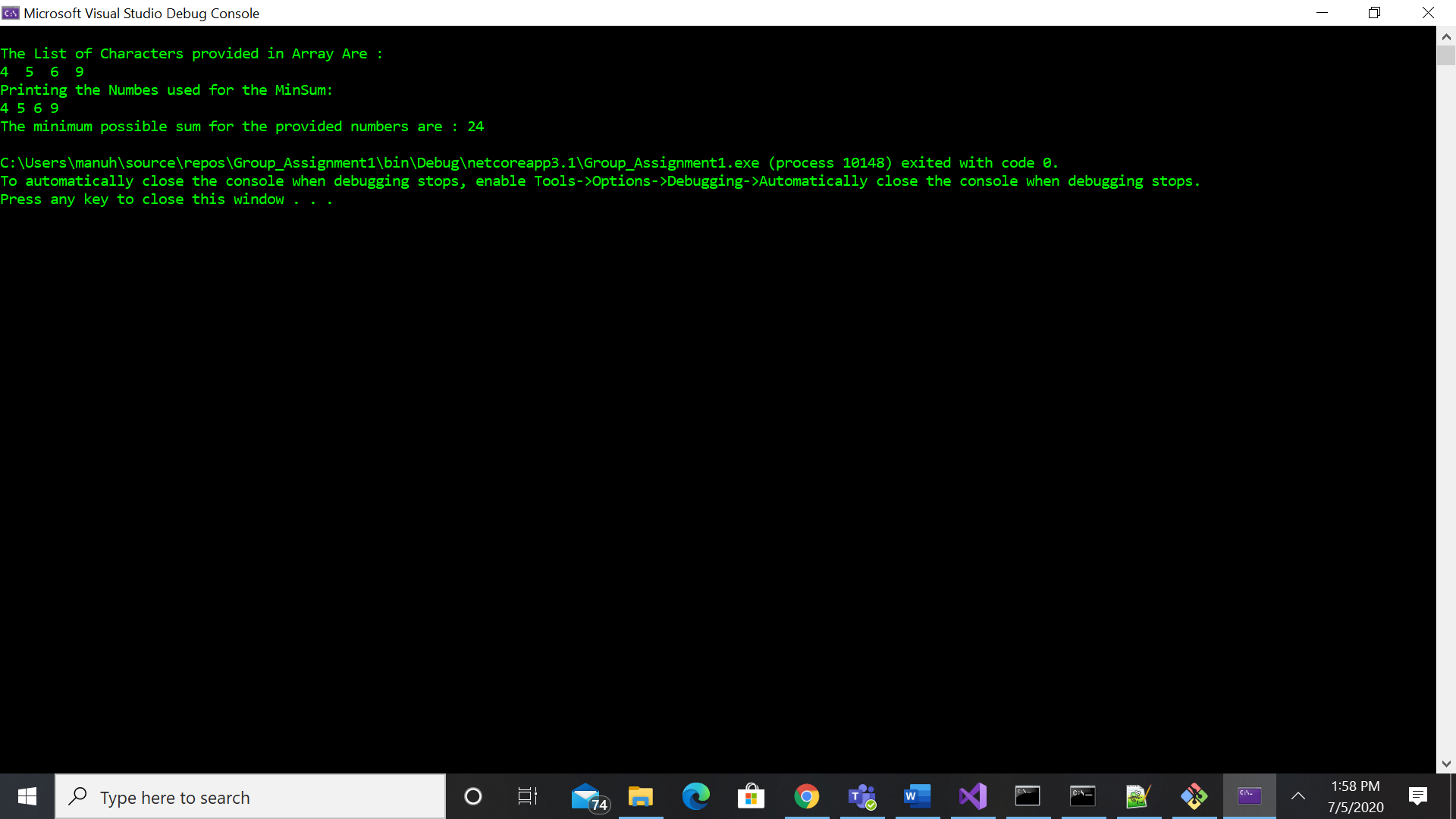


**Example-2:**





**Example-3:**



## Question 4:

You are given a string and your task is to sort the given string in decreasing order of frequency of occurrence of each character.

Example 1:

Input : “Dell” Output: “llDe”

Explanation: “lleD” is also accepted. Frequency of l is 2 and other 2 characters is 1.

Example 2:

Input : eebhhh Output: hhheeb

Example 3:

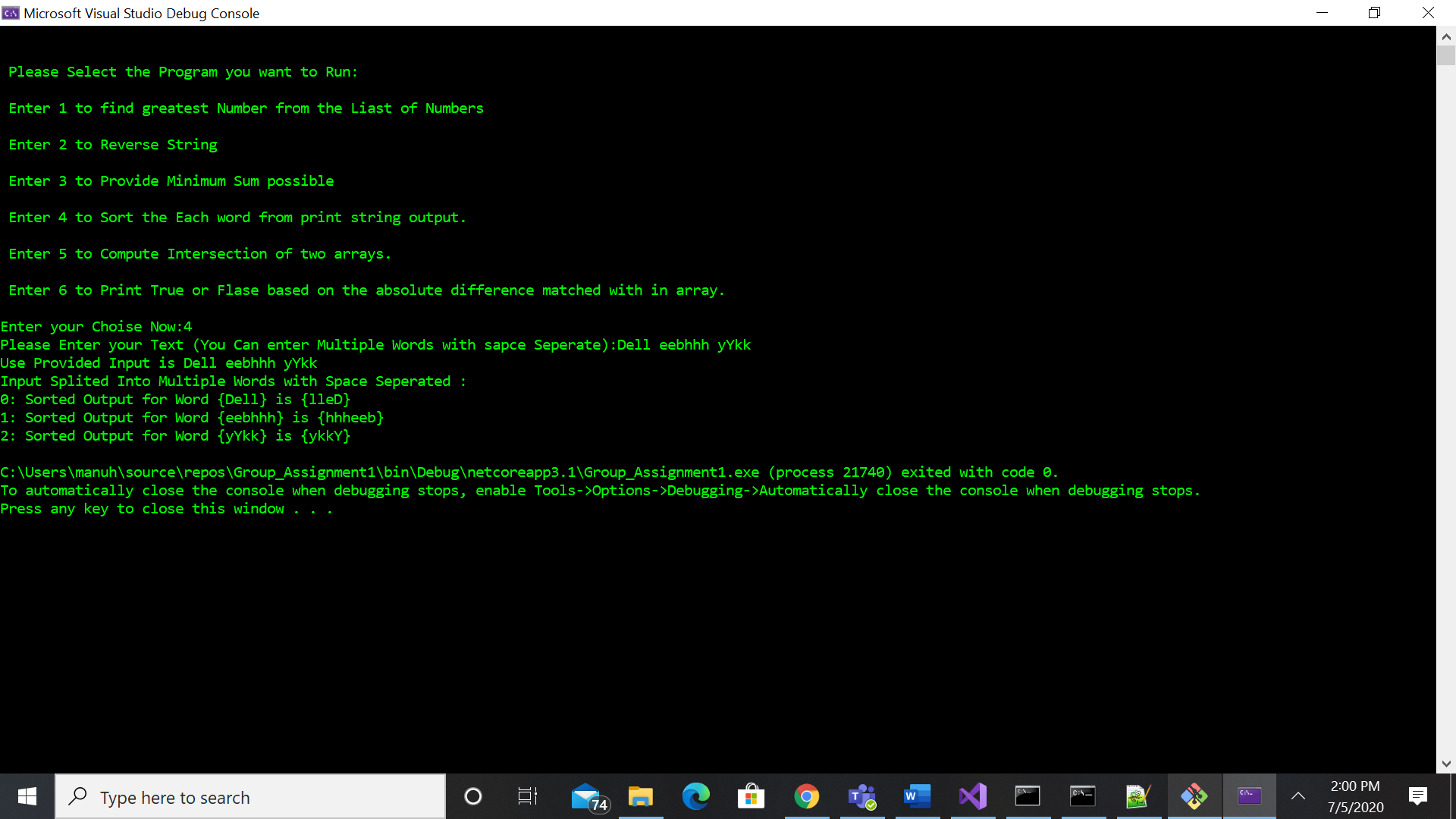
Input : yYkk Output: kkYy

Explanation: “Yykk” is not a valid answer as Y and y are treated as two different characters.

**Note: The solution must use a dictionary as the primary data structure.**

**public static string FreqSort(string s)**

**Program Output: For this Program we can either Give Multiple Words as input String or Individual Words**



## Question 5:

Rocky the Bull is new to programming and is having trouble understating the importance of time complexity. Professor Agrawal assigned you the job of explaining time complexity to Rocky with the example below.

Given two arrays, write a function to compute their intersection.

Example 1:

Input: nums1 = [2,5,5,2], nums2 = [5,5]

Output: [5,5]

Example 2:

Input: nums1 = [3,6,2], nums2 = [6,3,6,7,3]

Output: [3,6]

**Note:**

* Each element in the result should appear as many times as it shows in both arrays.
* The result can be in any order.
* Submit 2 Solutions where time complexity of the methods is be **less than O(n^2)**.

**Hint1:** You can sort the both arrays and then compute the intersection, in this way you will achieve a time complexity of **O(n log n)**.

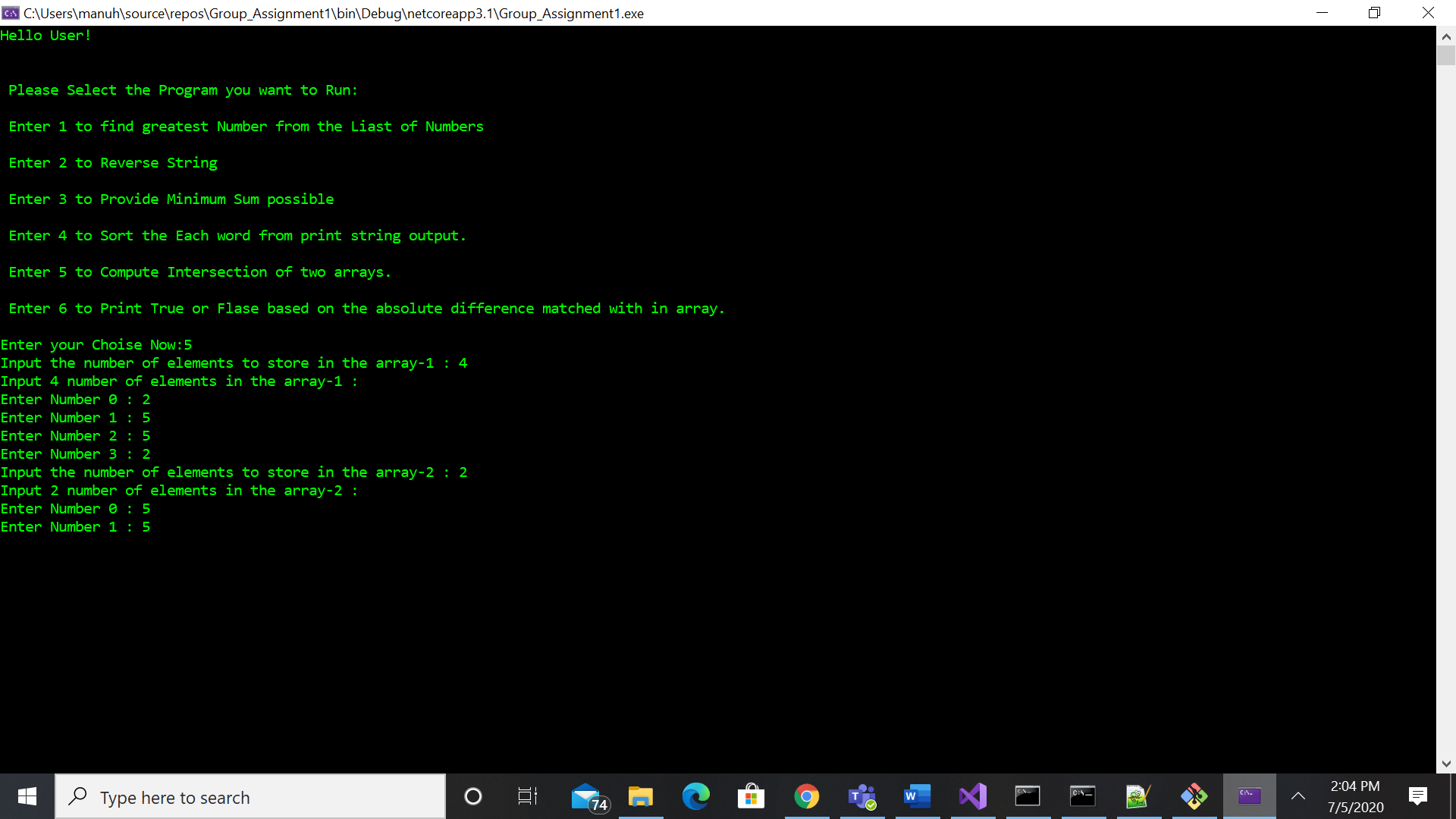
**Hint2:** If you make use of a dictionary, you need not sort the arrays to compute the intersection. The time complexity in this case will be **O(n)**.

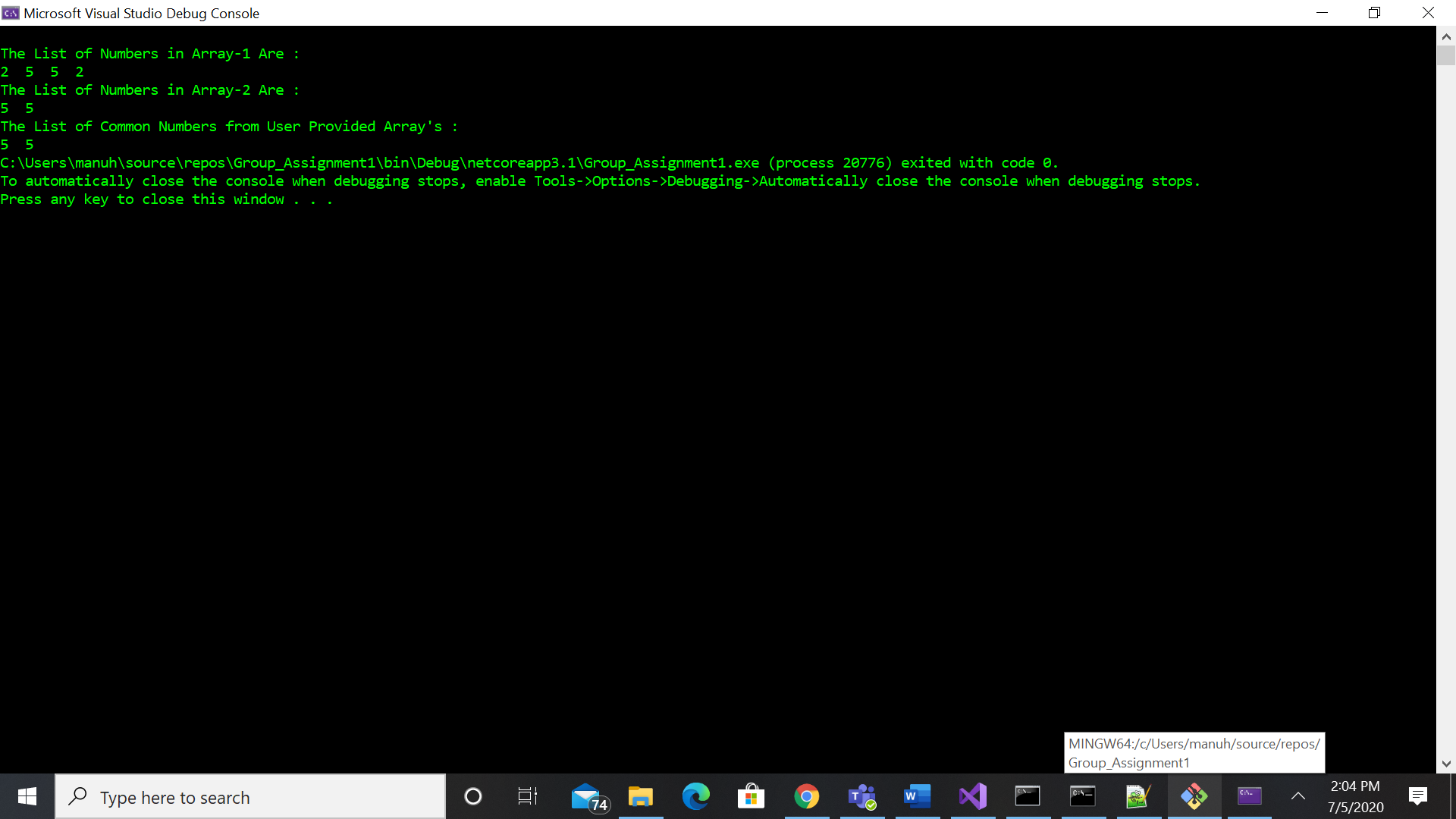
**public static int[] Intersect1(int[] nums1, int[] nums2)**

**public static int[] Intersect1(int[] nums1, int[] nums2)**

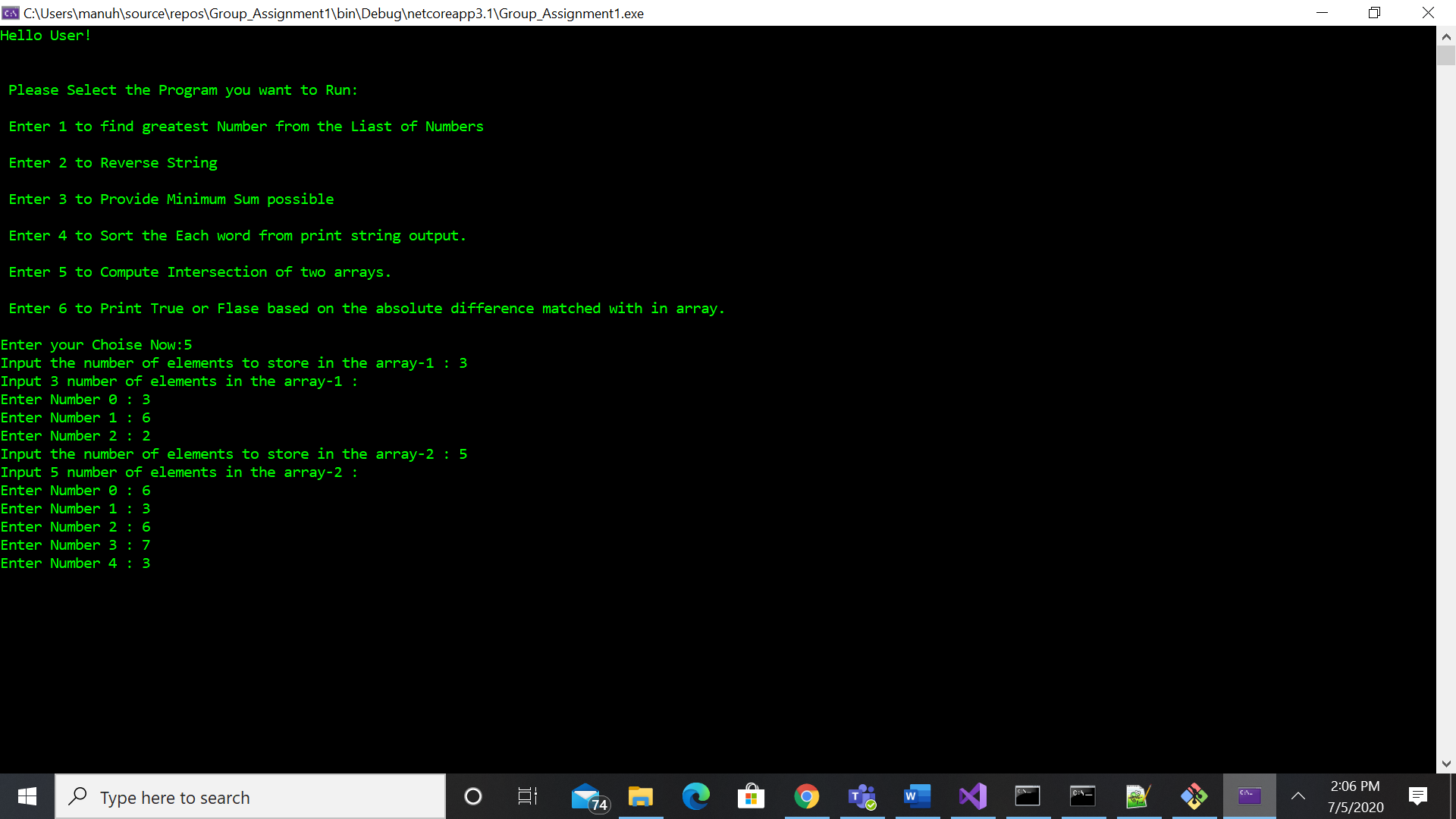
**Program Output: Please find the program output for the multiple examples shared above.**

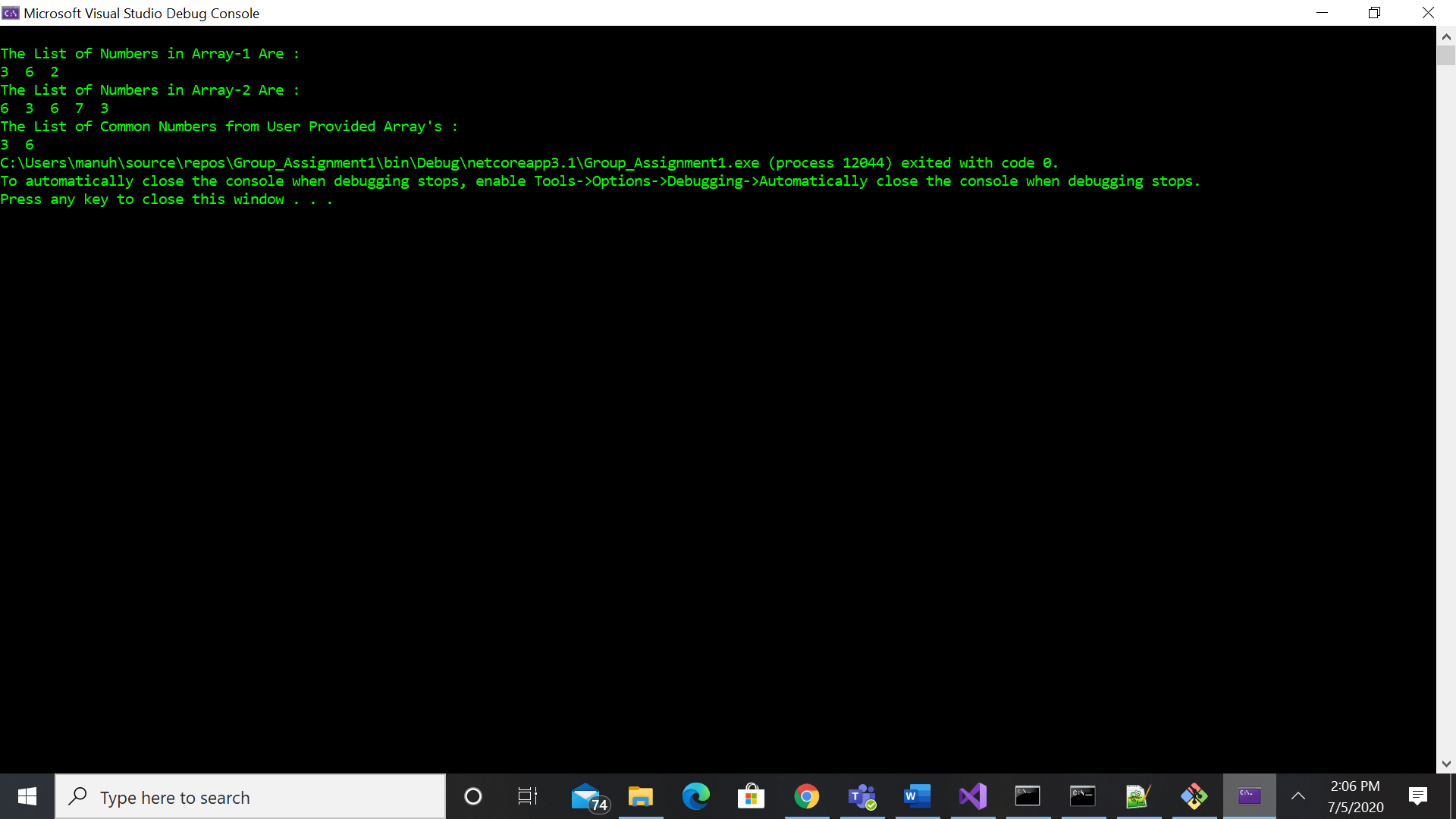
**Example-1:**





**Example-2:**





## Question 6:

You are given an array of characters and an integer k, and are required to find out whether there are two distinct indices i and j in the array such that arr[i]=arr[j] and the absolute difference between i and j is at most k.

Example 1:

Input: arr=[a,g,h,a], k=3

Output= true

Explanation= arr[0] and arr[3] contains “a”; hence the absolute difference between 0 and 3 is 3 which is equal to 3 therefore output is true.

Example 2: arr=[k,y,k,k], k=1

Output= true

Example 3:

Input: arr=[a,b,c,a,b,c], k=2

Output: false

Explanation= arr[0]=arr[3] but absolute difference is not at most 2

arr[1]=arr[4] but absolute difference is not at most 2

arr[2]=arr[5] but absolute difference is not at most 2

There fore output is false.

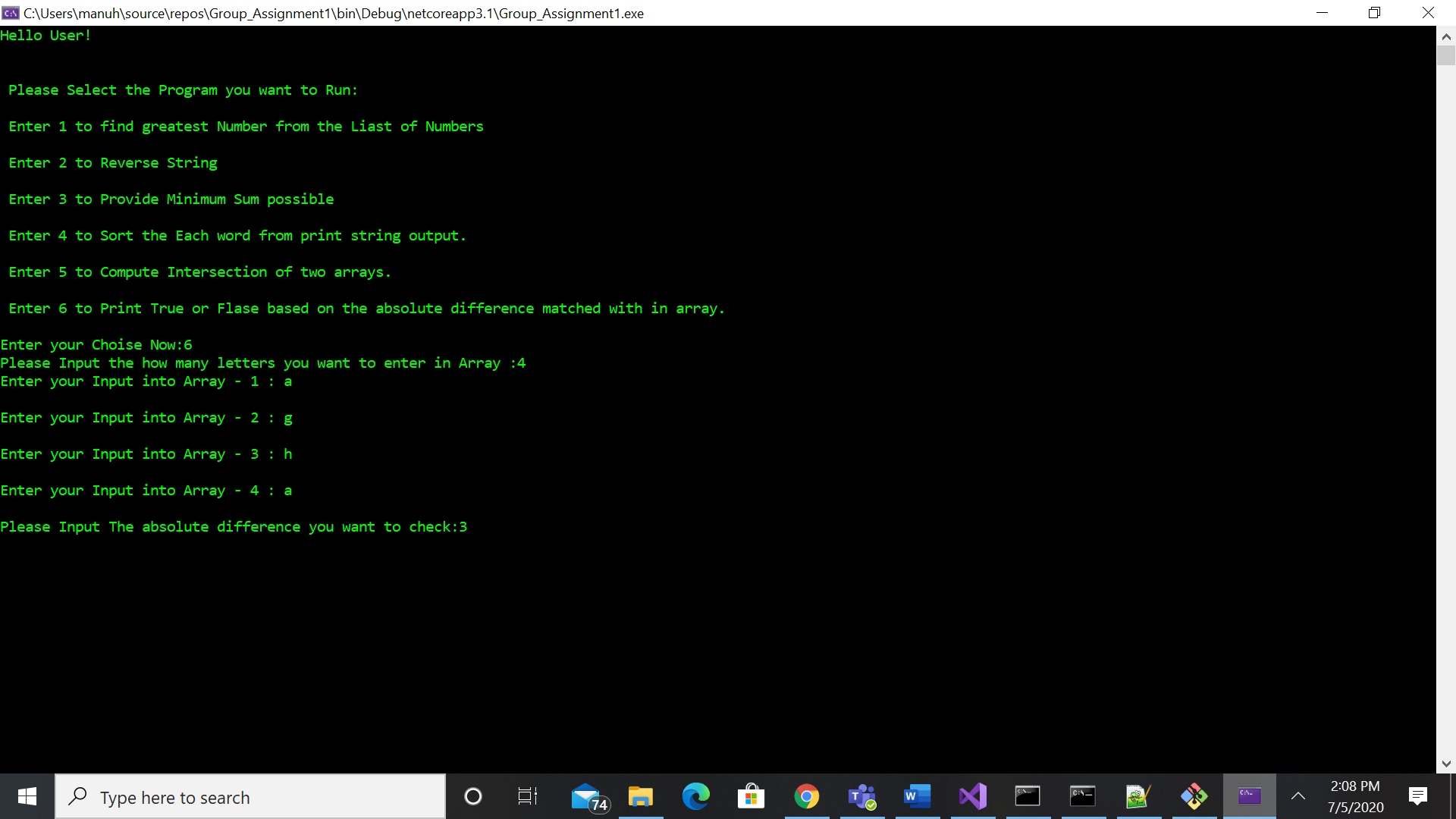
Note: The algorithm’s runtime complexity must be in the order of O(n).

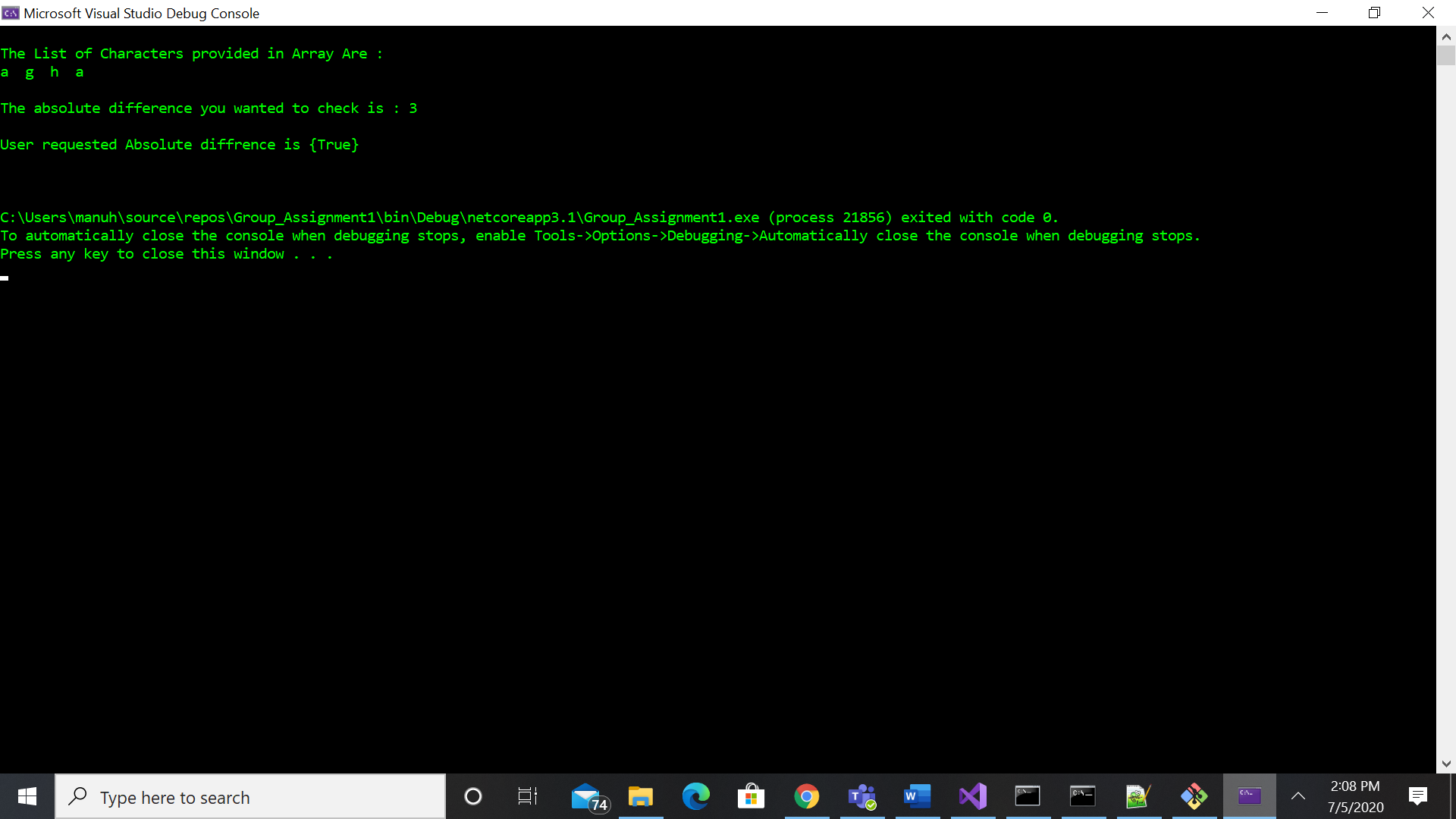
Hint : You can make use of a dictionary to obtain the required time complexity.

**public bool ContainsDuplicate(char[] arr, int k)**

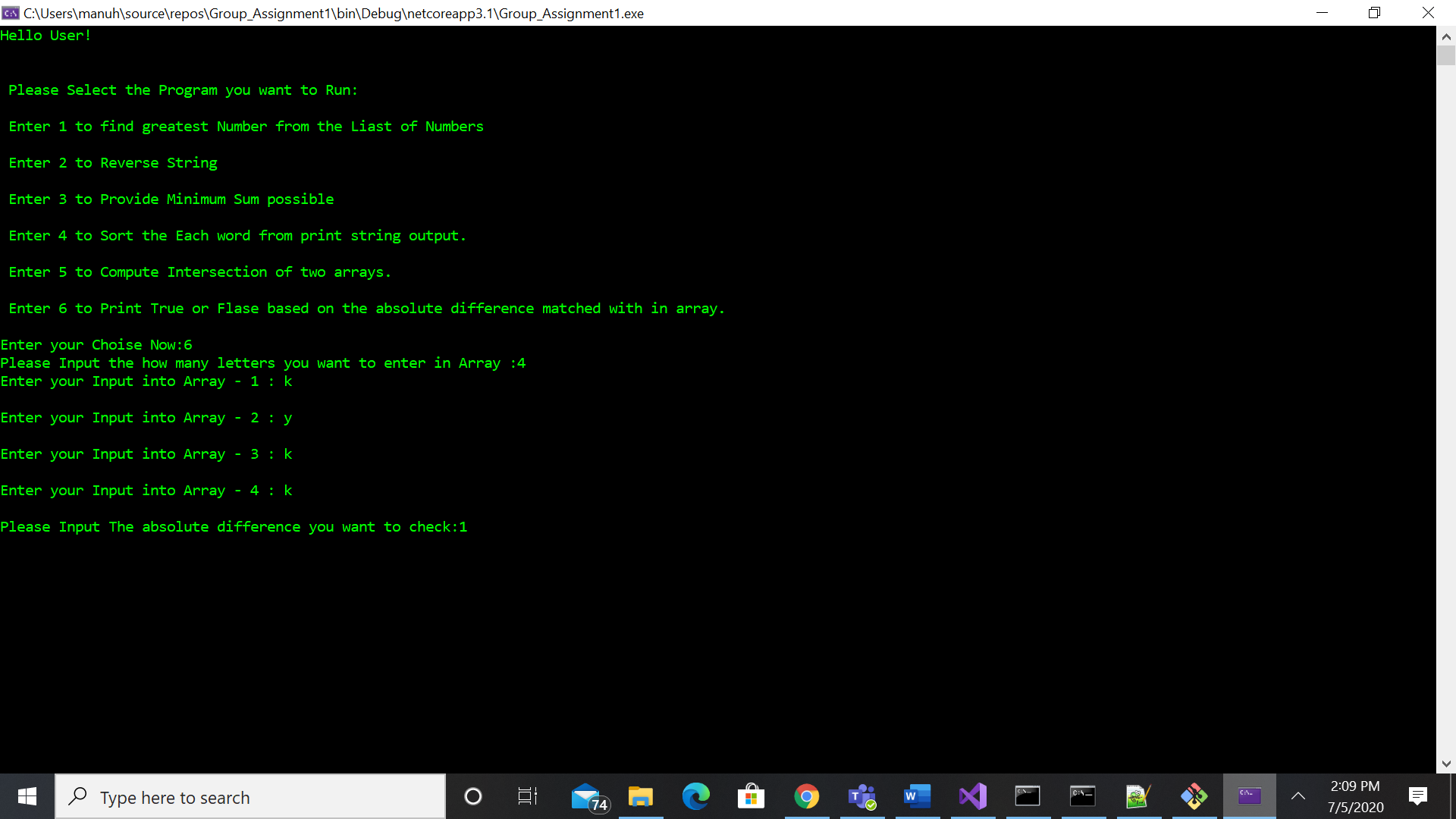
**Program Output: Please find the program output for the multiple examples shared above.**

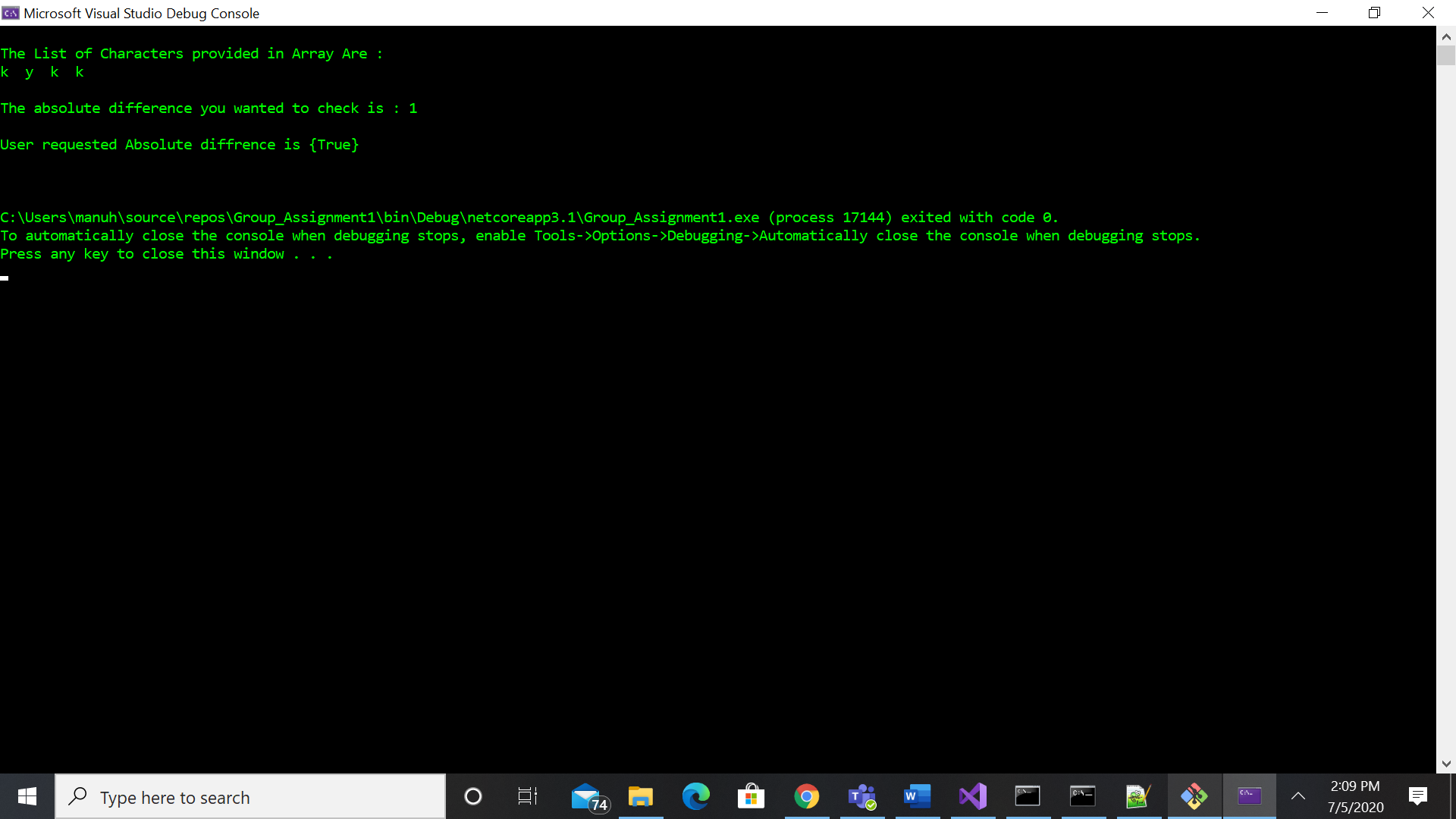
Example-1:





Example-2:





Example-3:

