NAME:MANVITH BALAJI

SECTION:A

MIS NO. :112315115

1. Write a program to count the numbers of characters in the given string and store them in a

dictionary data structure.

Input code:

a=input("Enter a string: ")

b=[]

c={}

for i in a:

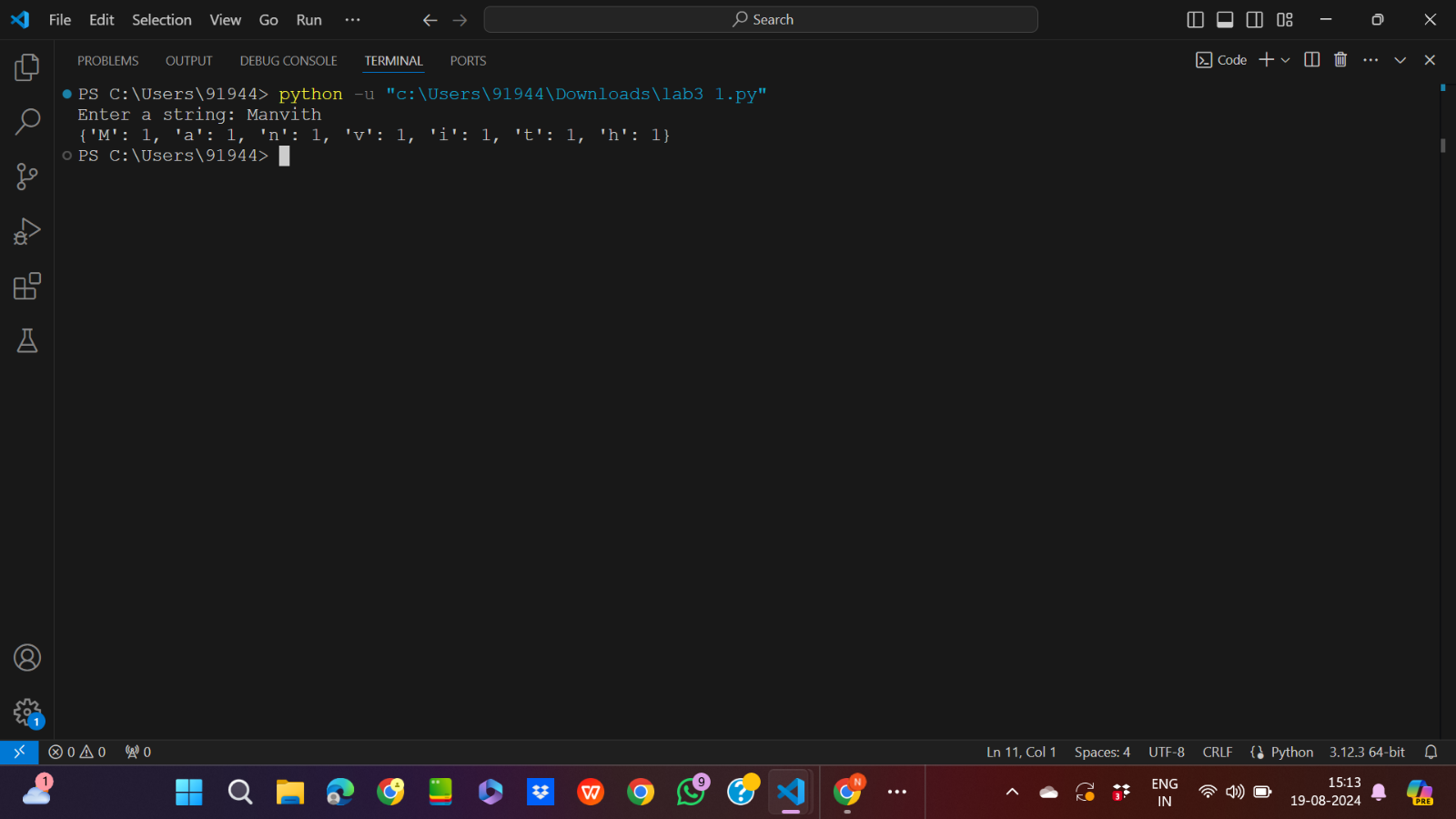
    if i not in b:

        c.update({i:a.count(i)})

        b.append(i)

print(c)

Output:



2. Write a program to use split and join methods in the given string and trace a birthday with a

dictionary data structure.

Input code:

a=input("Enter a string: ")

dob=input("Enter your date of birth (dd-mm-yy): ")

c={}

list1=a.split(" ")

print("The list form of string is: ",list1)

b="\*".join(a)

print(f"The string after join method is {b}")

list2=dob.split("-")

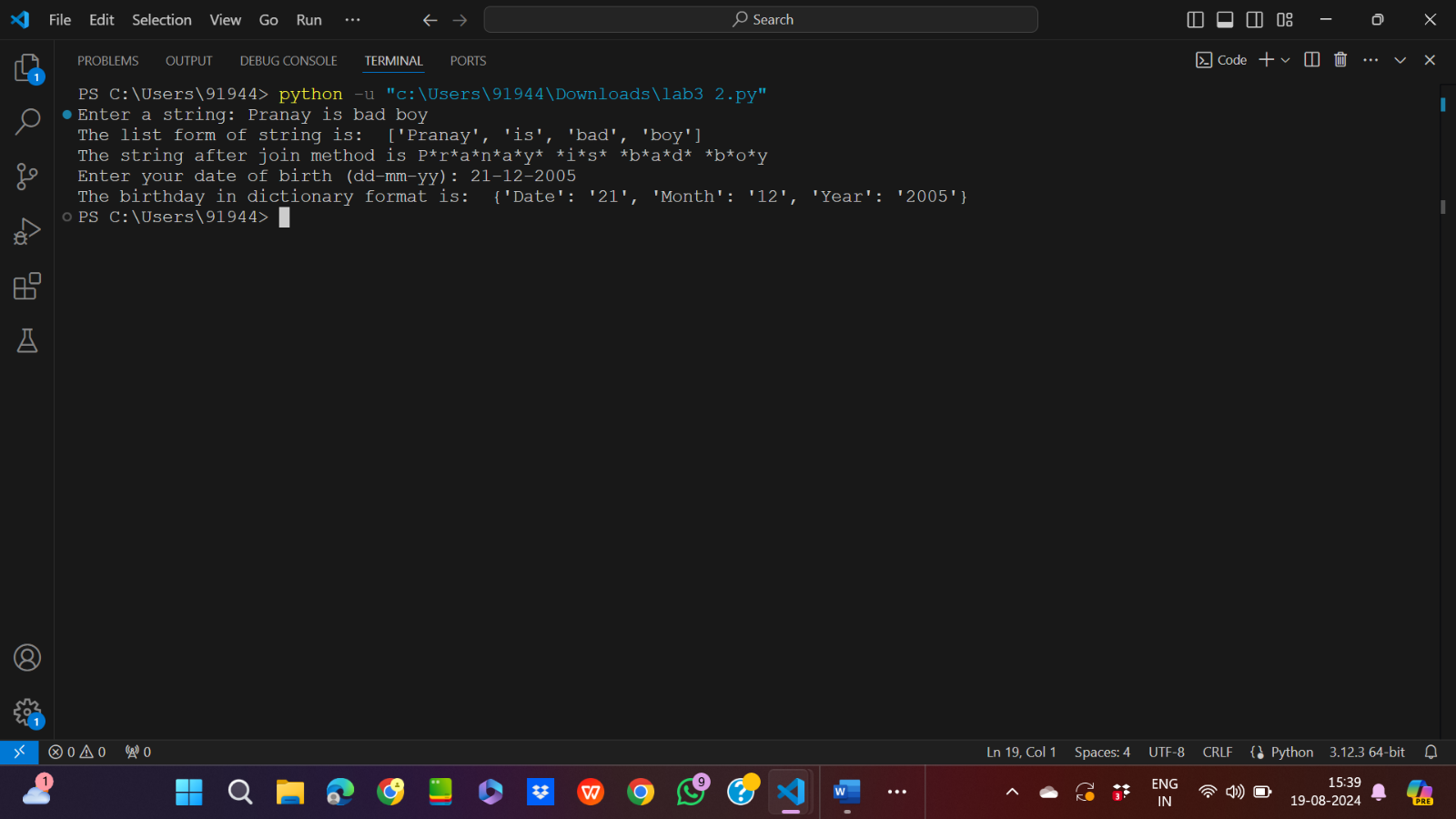
list3=["Date","Month","Year"]

for i in range(3):

    c.update({list3[i]:list2[i]})

print("The birthday in dictionary format is: ",c)

Output:



3. Write function to compute gcd and lcm of two numbers.

Input code:

from math import \*

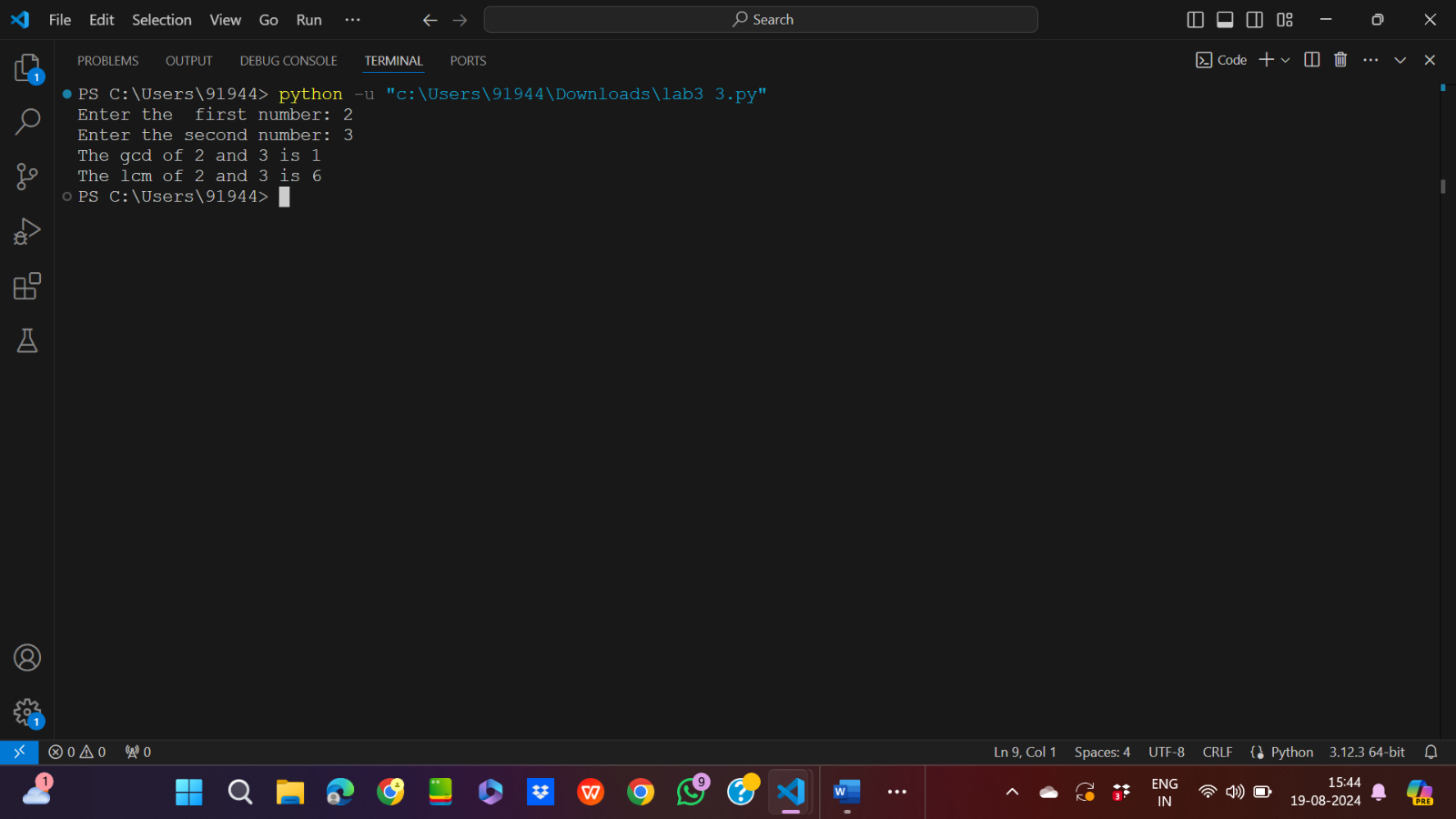
num1=int(input("Enter the  first number: "))

num2=int(input("Enter the second number: "))

print(f"The gcd of {num1} and {num2} is {gcd(num1,num2)}")

print(f"The lcm of {num1} and {num2} is {lcm(num1,num2)}")

Output:



4. Write a function ball \_collide that takes two balls as parameters and computes if they are

colliding. Your function should return a Boolean representing whether or not the balls are

colliding. Represent a ball on a plane as a tuple of (x, y, r), r being the radius. If (distance

between two balls centers) &lt;= (sum of their radii) then (they are colliding)

Input code:

from math import sqrt

def ball\_collide(b1,b2):

    if sqrt(((b2[0]-b1[0])\*\*2+(b2[1]-b1[1])\*\*2)<=b1[0]+b2[0]):

        return True

    else:

        return False

x1,y1,r1=int(input("Enter the x-coordinte of ball1: ")),int(input("Enter the y-coordinte of ball1: ")),int(input("Enter the radiusof ball1: "))

x2,y2,r2=int(input("Enter the x-coordinte of ball2: ")),int(input("Enter the y-coordinte of ball2: ")),int(input("Enter the radius of ball2: "))

a=ball\_collide((x1,y1,r1),(x2,y2,r2))

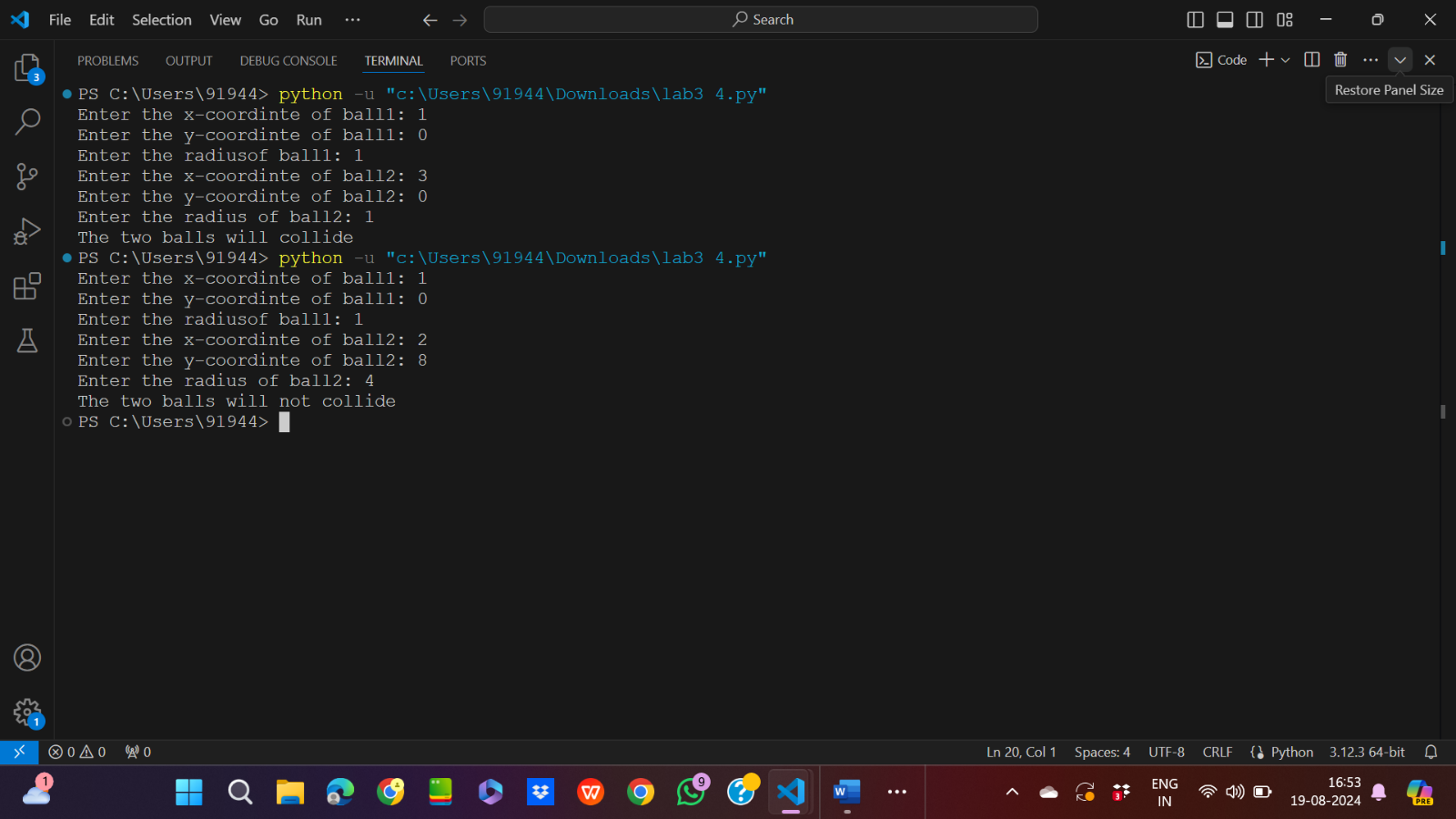
if(a==True):

    print("The two balls will collide")

else:

    print("The two balls will not collide")

Output:



5. Find mean, median, mode for the given set of numbers in a list.

Input code:

num=[2,3,6,7,8,9,9,0,8,7,7]

b=[]

max=[]

sum=0

for i in num:

    sum=sum+num[i]

print(f"The mean of the numbers is {sum/len(num)}")

for i in num:

    if i not in b:

        max.append(num.count(i))

        b.append(i)

max.sort(reverse=True)

for j in num:

    if j not in b:

        b.append(j)

for j in b:

    if max[0]==num.count(j):

        print(f"The mode of these numbers is {j}")

num.sort()

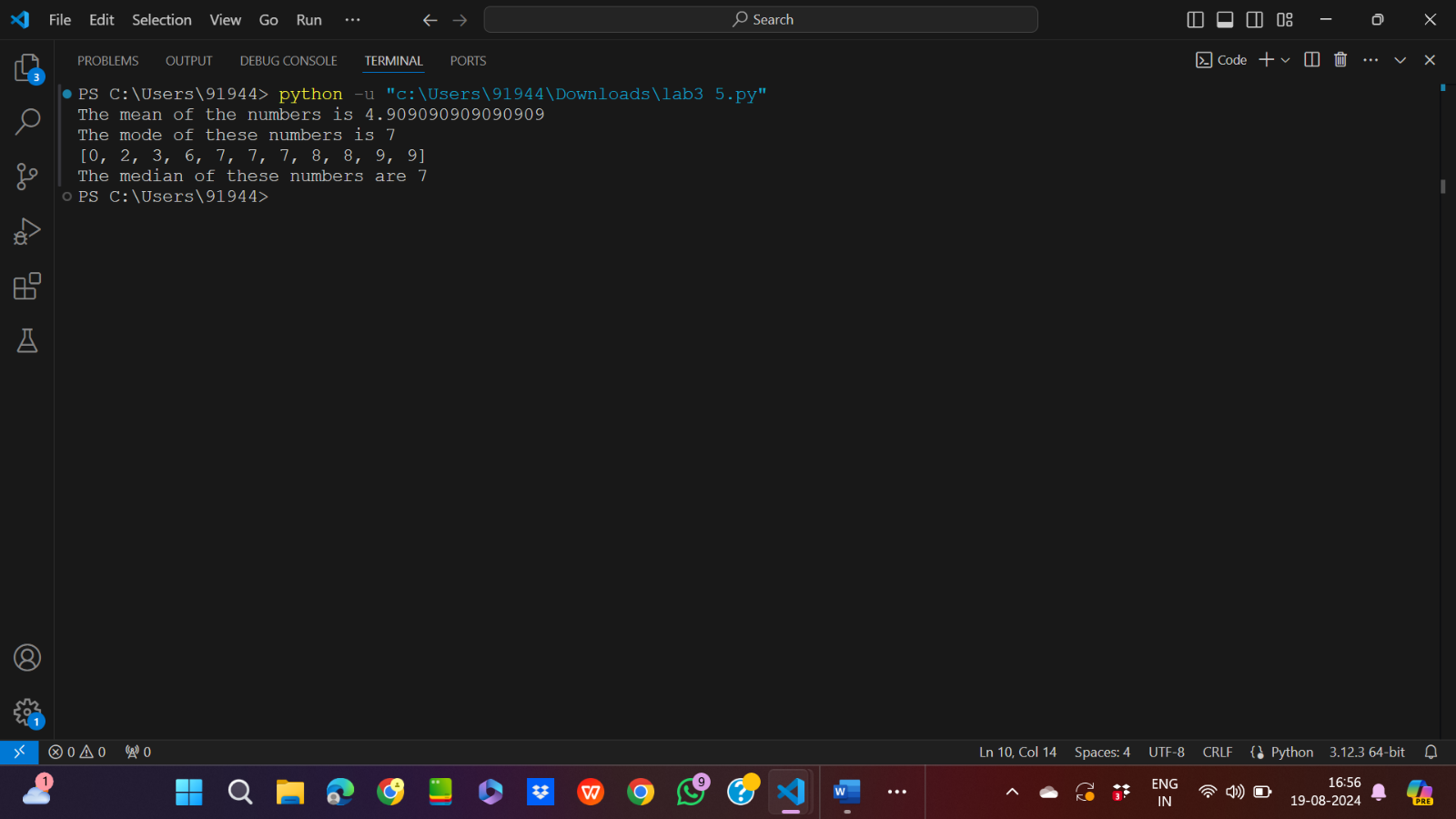
if(len(num)%2==0):

    print(f"The median of these numbers are {num[len(num)//2-1]+num[len(num)//2]}")

else:

   print(f"The median of these numbers are {num[len(num)//2]}")

Output:



6. 6. Write a program to implement

a. Bubble sort,

b. Merge sort,

c. Selection sort and

d. Insertion sort.

Execute these sorting algorithms using switch case.

Input code:

a.

num=[3,4,56,78,8,9,0,8,8,8,9]

for i in range(1,len(num)):

    for j in range(1,len(num)-i+1):

        match num[j-1]>num[j]:

            case True:

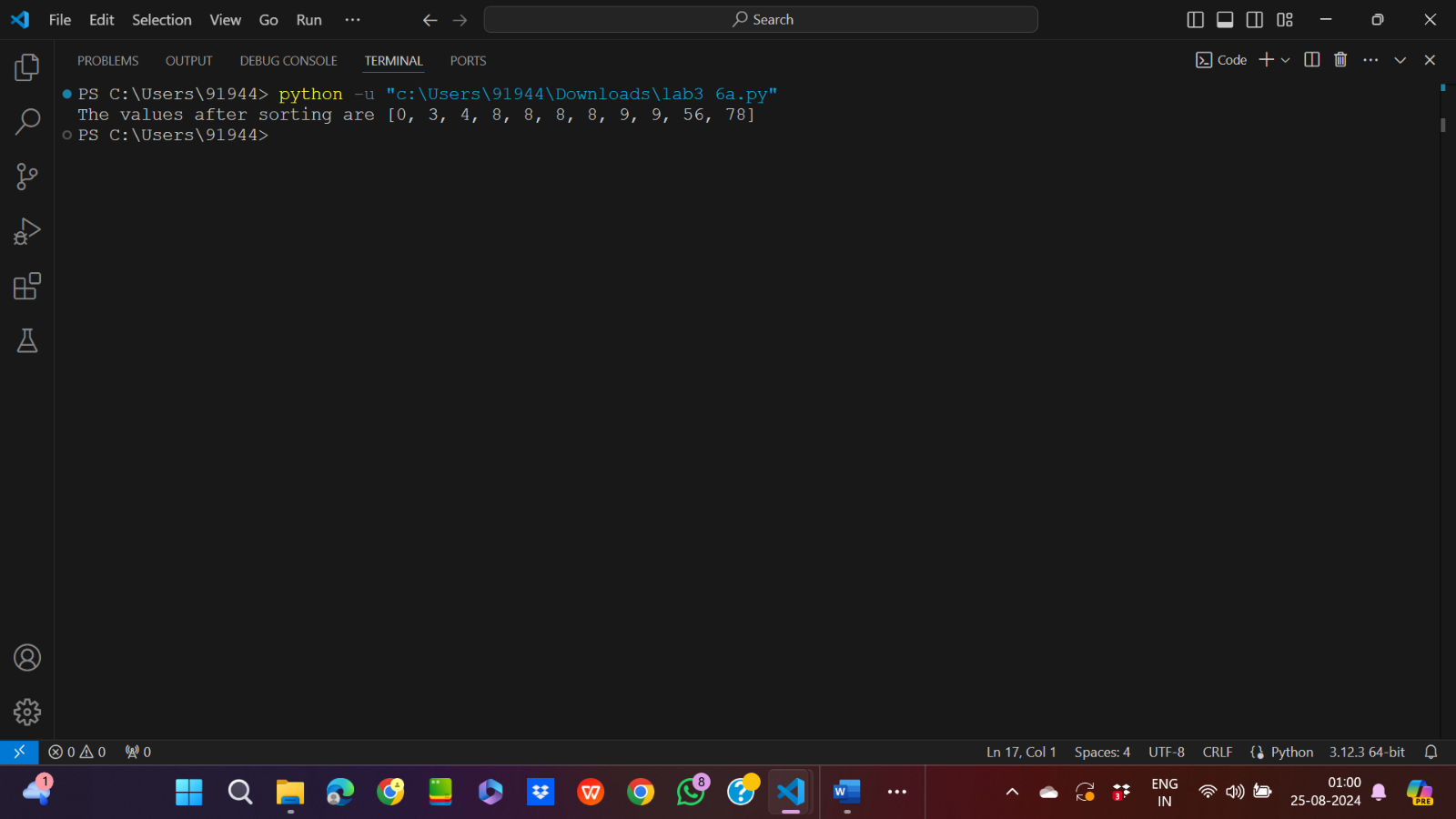
                num[j-1],num[j]=num[j],num[j-1]

            case False:

                continue

print(f"The values after sorting are {num}")

Output:



b.Merge Sort

Input code:

num=[2,3,5,7,8,9,9,77,6,7,7,88,8,8]

def merge(left, right):

    result = []

    match bool(left and right):

        case True:

            while left and right:

                if left[0] <= right[0]:

                    result.append(left.pop(0))

                else:

                    result.append(right.pop(0))

            return result+right+left

def merge\_sort(arr):

    if len(arr) == 1:

        return arr

    mid = len(arr) // 2

    left = merge\_sort(arr[:mid])

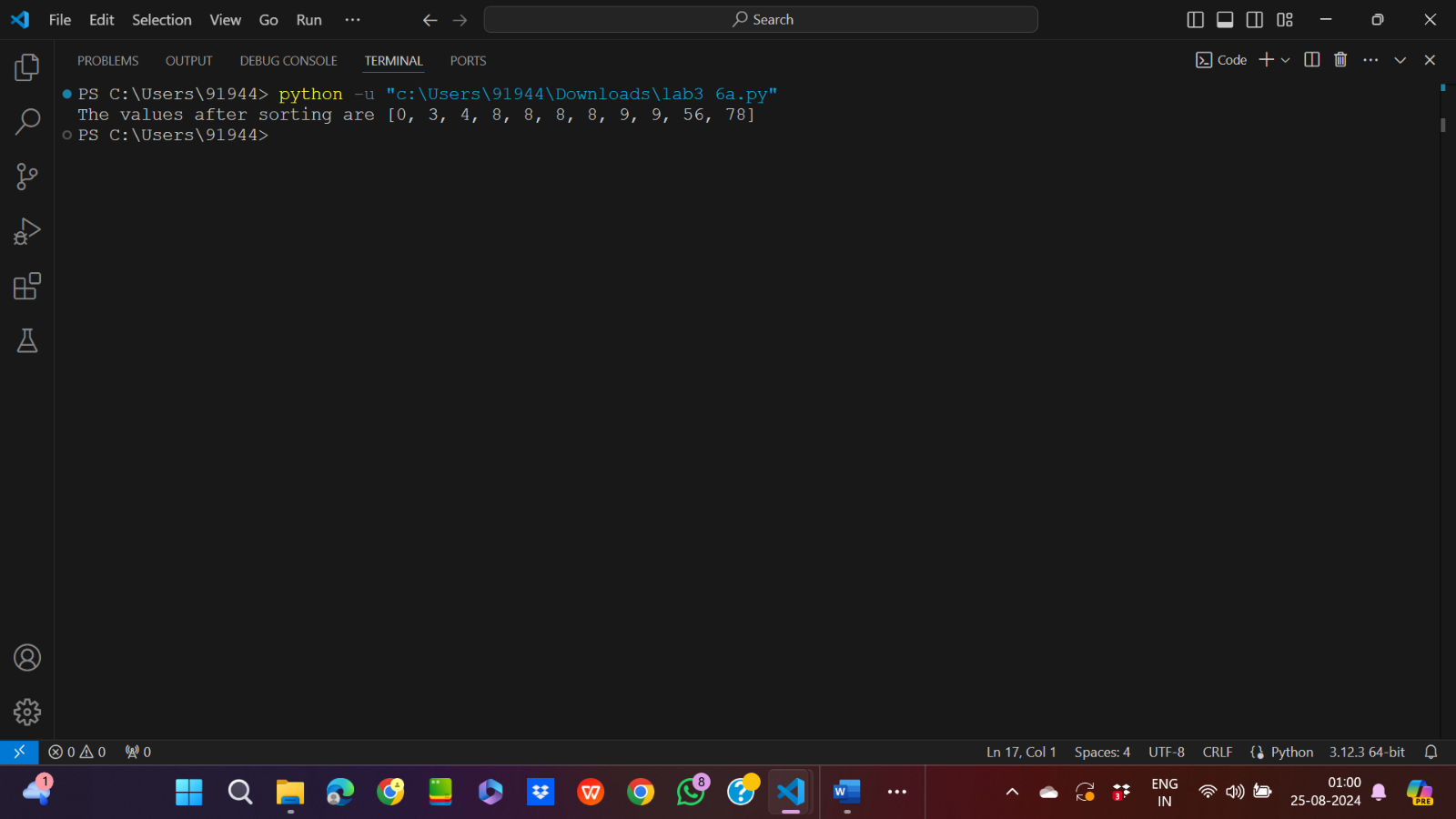
    right = merge\_sort(arr[mid:])

    return merge(left, right)

c=merge\_sort(num)

print(c)

Output:



c.

Selection Sort

Input code:

num=[2,3,8,9,7,6,66,5,5,5,7,8,9,7]

for i in range(0,len(num)-1):

    min=i

    for j in range(i+1,len(num)):

        if num[min]>num[j]:

            min=j

    match min!=i:

        case True:

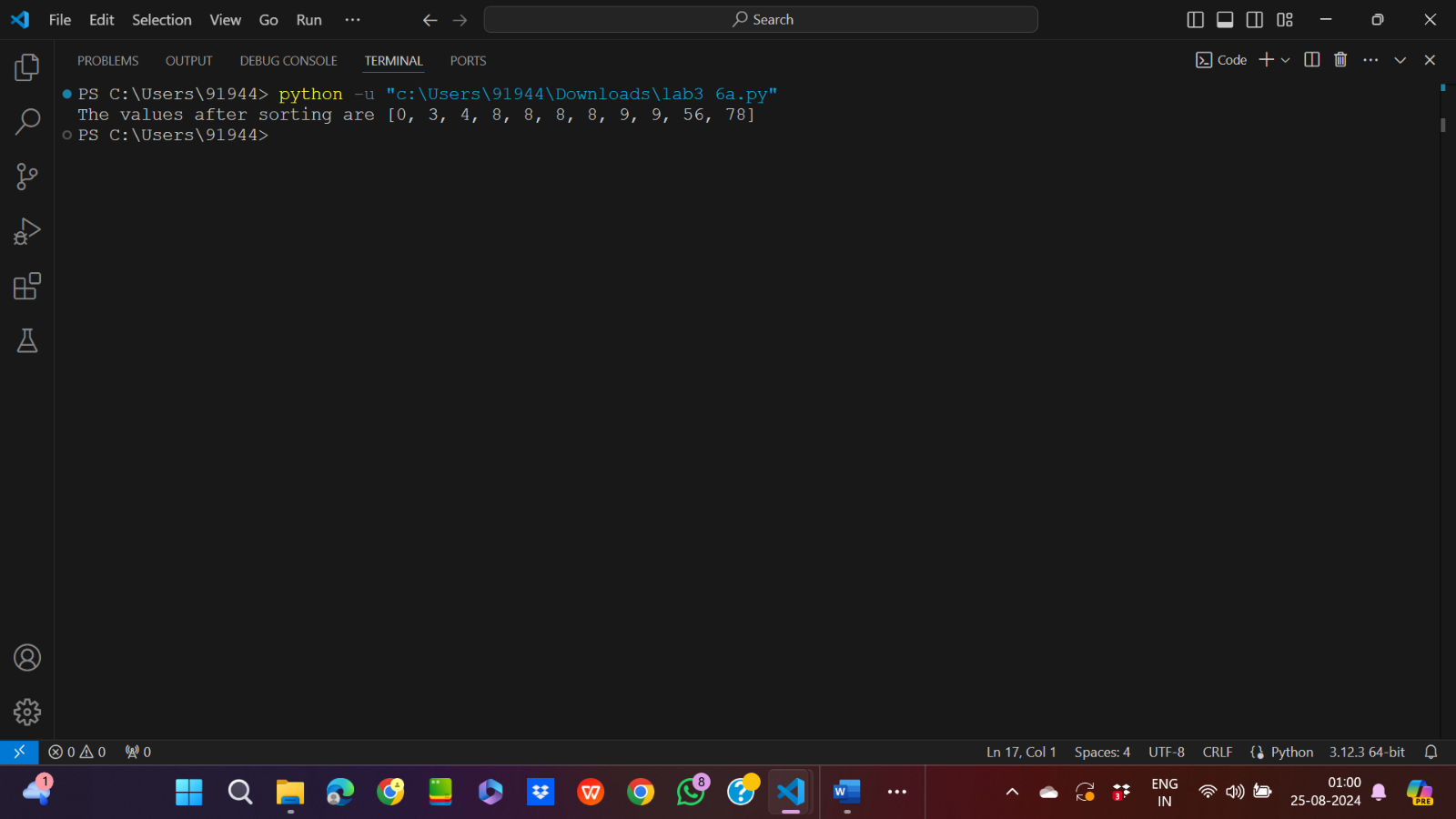
            num[min],num[i]=num[i],num[min]

        case False:

            continue

print(num)

Output:



d.Insertion Sort

Input code:

num=[2,4,56,7,8,9,6,5,4,4,5,7,8]

for i in range(1,len(num)):

    key=num[i]

    j=i-1

    match j>=0:

        case True:

            while(j>=0 and num[j]>key):

                num[j+1]=num[j]

                j=j-1

        case False:

            continue

    num[j+1]=key

print(num)

Output:

