Roll No : 58

Name : Ashwini Arun Mahajan

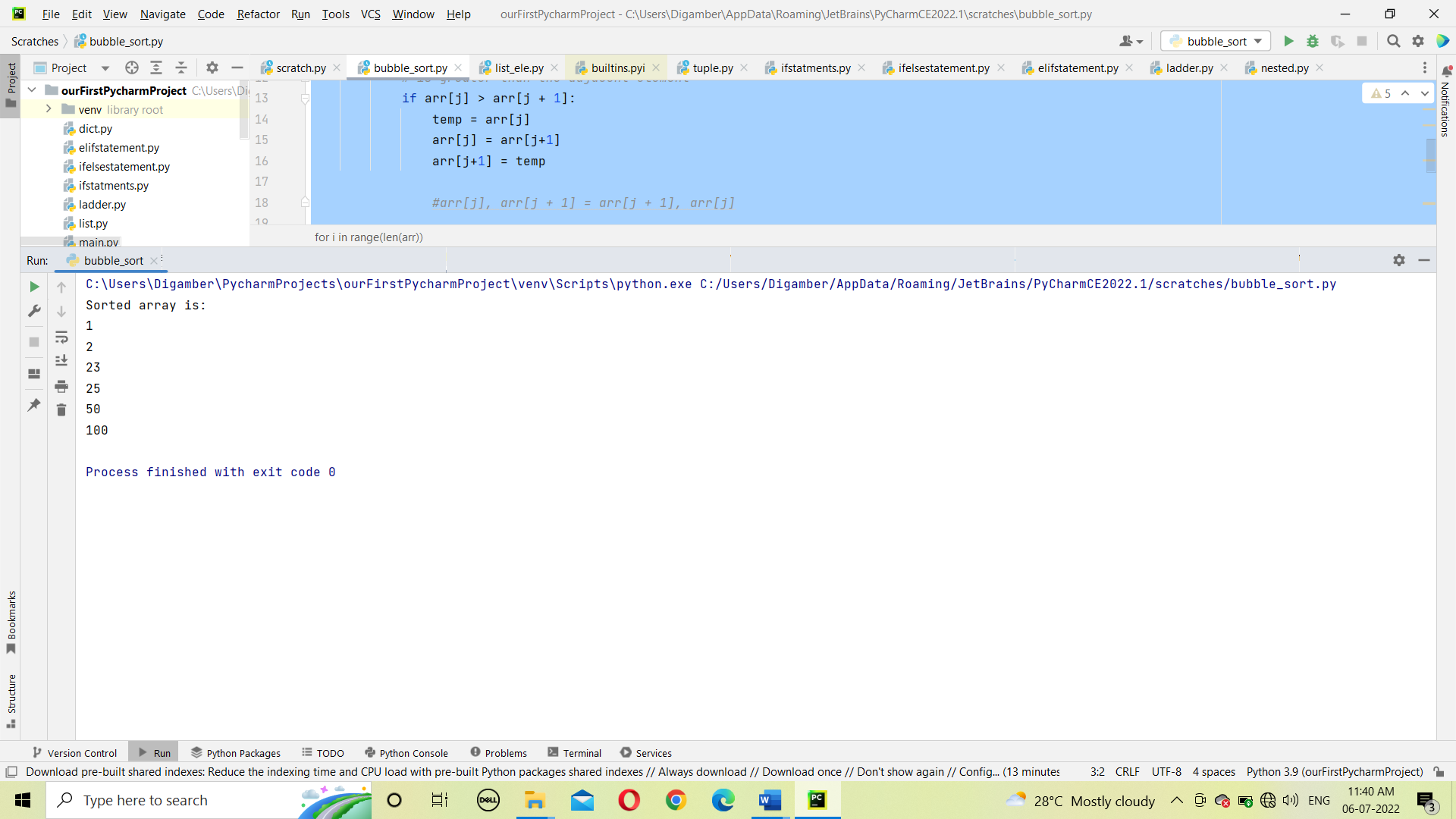
Assignment No : 1.5

Assignment Name : Develop programs for data structure algorithms using python – searching, sorting and hash tables.

**4. Write a python programme for use of bubble sort algorithm to sort the given list.**

def bubble\_Sort(arr):  
 n = len(arr)  
  
for i in range(n):  
 for j in range(0, n - i - 1):  
  
 if arr[j] > arr[j + 1]:  
 temp = arr[j]  
 arr[j] = arr[j+1]  
 arr[j+1] = temp  
  
 arr = [2, 1, 100, 23, 25, 50]  
  
bubbleSort(arr)  
  
print("Sorted array is:")  
for i in range(len(arr)):  
 print("%d" % arr[i])

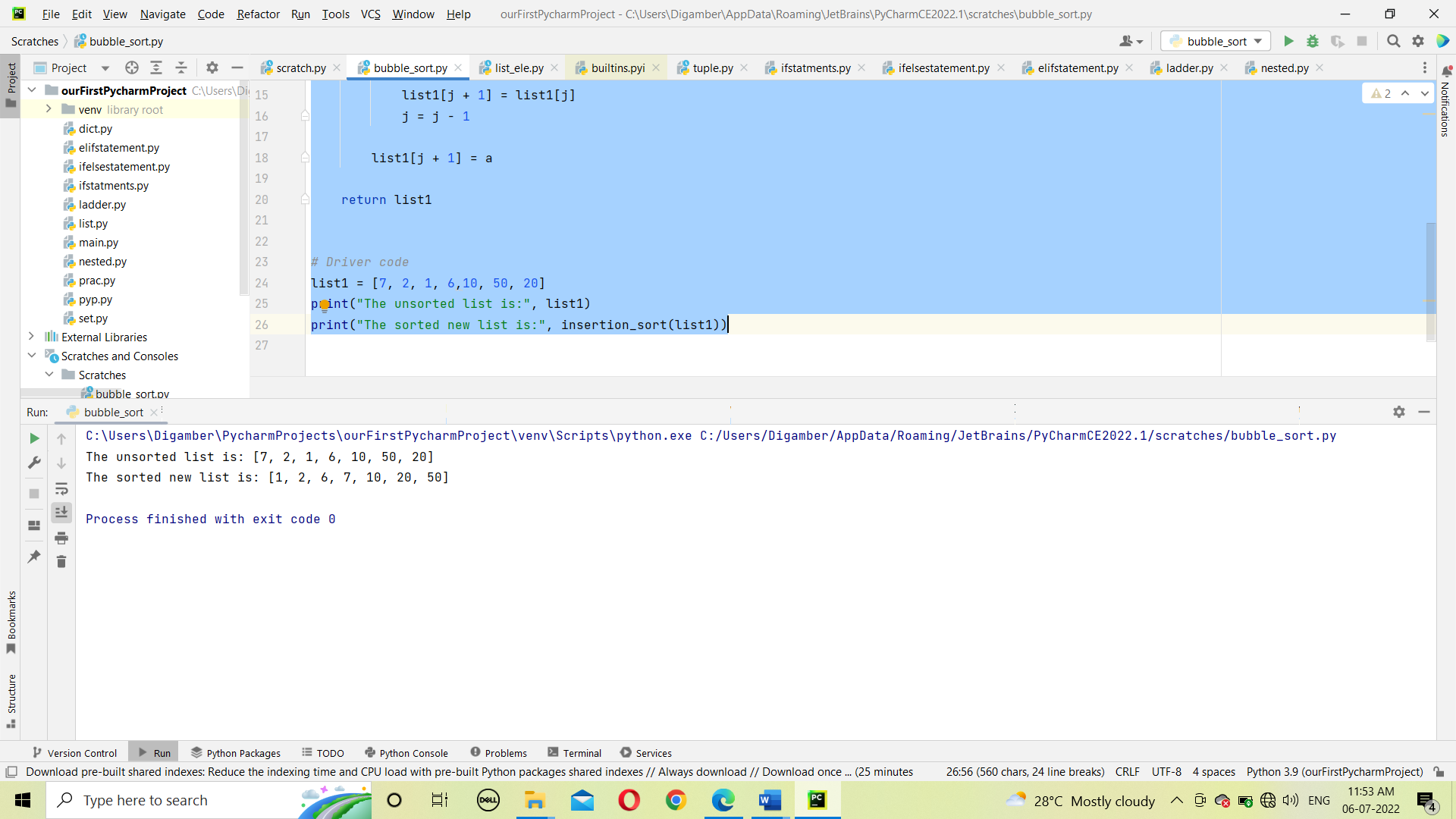
Output :



**6. Write a python programme for use of insertion sort algorithm to sort element in list.**

def insertion\_sort(list1):  
 for i in range(1, len(list1)):  
  
 a = list1[i]  
  
 j = i - 1  
  
 while j >= 0 and a < list1[j]:  
  
 list1[j + 1] = list1[j]  
 j = j - 1  
  
 list1[j + 1] = a  
  
 return list1  
  
list1 = [7, 2, 1, 6,10, 50, 20]  
print("The unsorted list is:", list1)  
print("The sorted new list is:", insertion\_sort(list1))

Output :

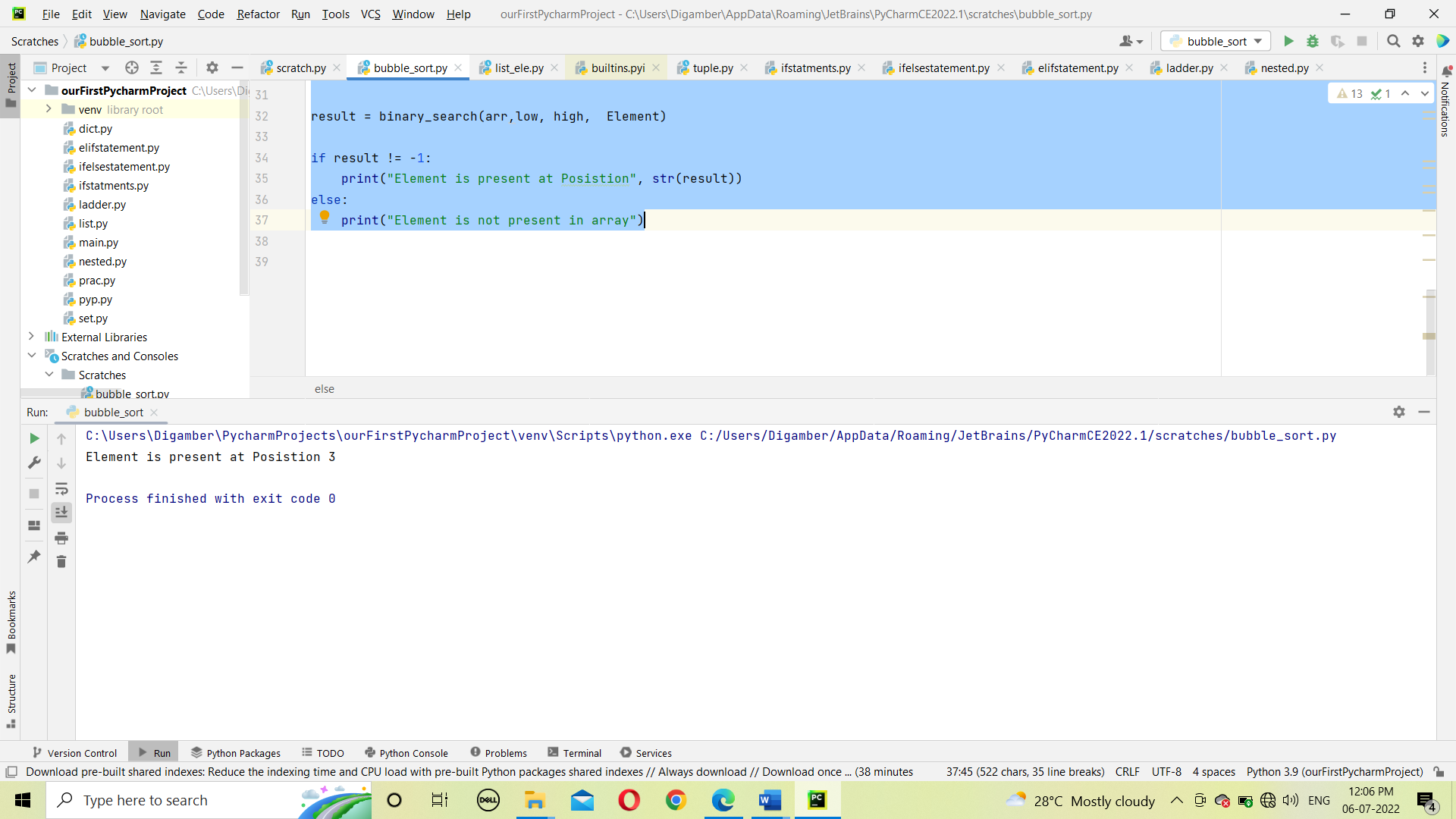


2. Write a python programme for use of Binary search algorithm to search element in

list.(Recursive)

def binary\_search(arr, low, high, ele):  
 mid = 0  
  
 while low <= high:  
  
 mid = (high + low) // 2  
  
   
 if arr[mid] < ele:  
 low = mid + 1  
  
   
 elif arr[mid] > ele:  
 high = mid - 1  
  
   
 else:  
 return mid+1  
  
   
 return -1  
  
  
  
arr = [2, 3, 4, 10, 40]  
Element = 4  
low = 0  
high = len(arr) - 1  
  
  
result = binary\_search(arr,low, high, Element)  
  
if result != -1:  
 print("Element is present at Posistion", str(result))  
else:  
 print("Element is not present in array")

Output :

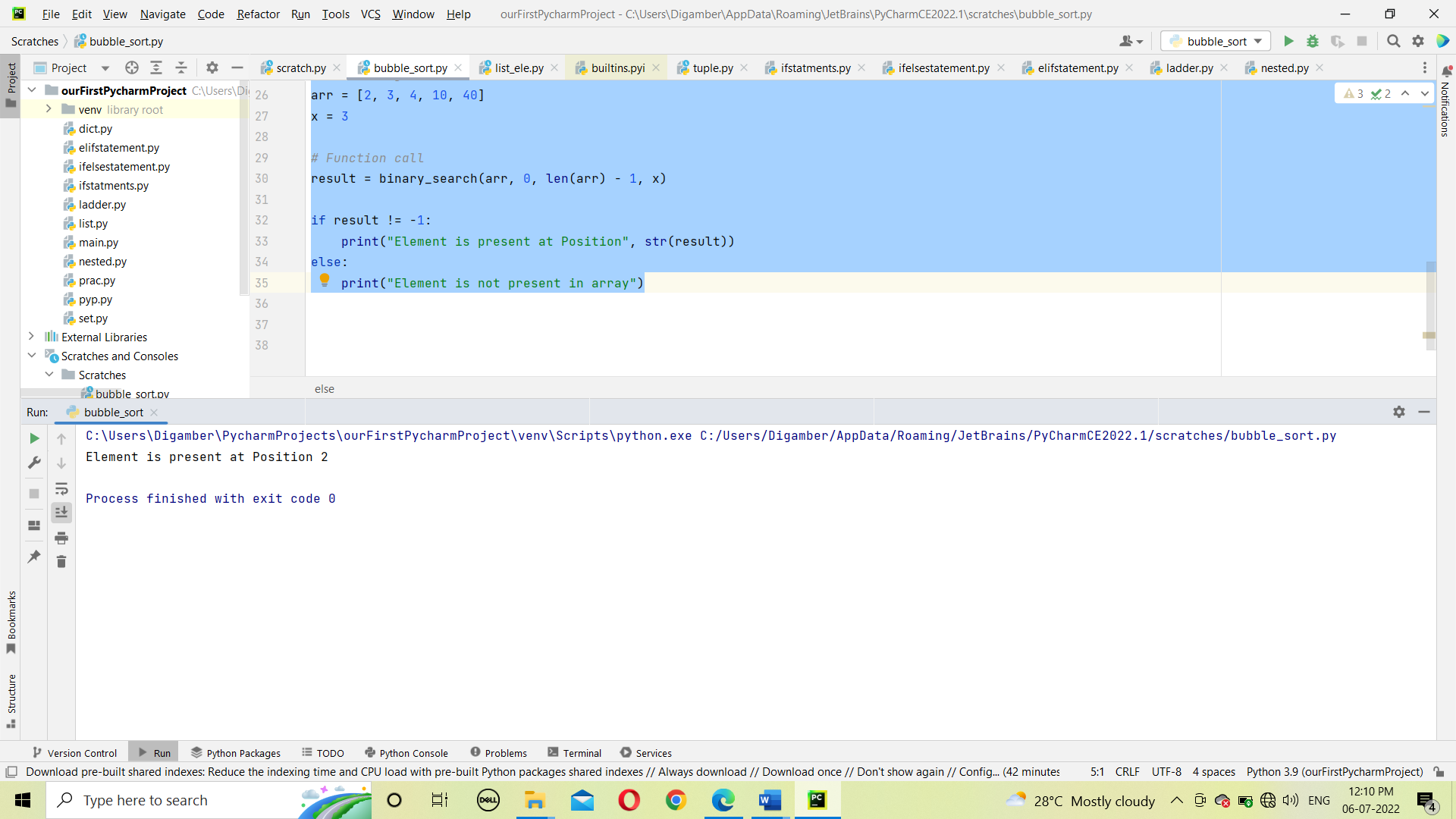


3. Write a python programme for use of Binary search algorithm to search element in \

list.(Without Using Recursion )

def binary\_search(arr, low, high, ele):  
 if high >= low:  
  
 mid = (high + low) // 2  
  
 *# If element is present at the middle itself* if arr[mid] == x:  
 return mid+1  
  
elif arr[mid] > x:  
 return binary\_search(arr, low, mid - 1, x)  
  
 else:  
 return binary\_search(arr, mid + 1, high, x)  
  
 else:  
 return -1  
  
  
arr = [2, 3, 4, 10, 40]  
x = 3  
  
result = binary\_search(arr, 0, len(arr) - 1, x)  
  
if result != -1:  
 print("Element is present at Position", str(result))  
else:  
 print("Element is not present in array")

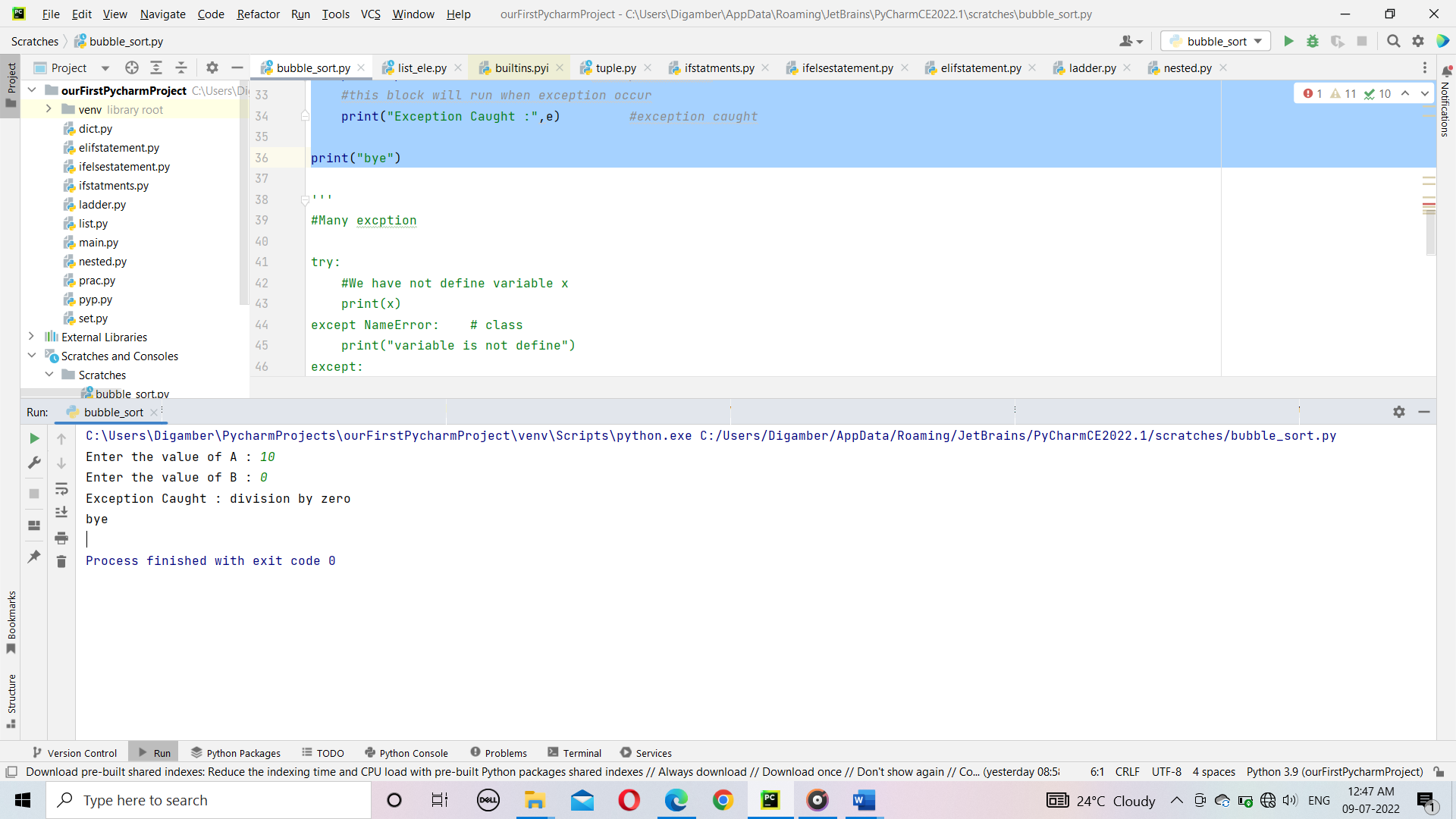
Output :

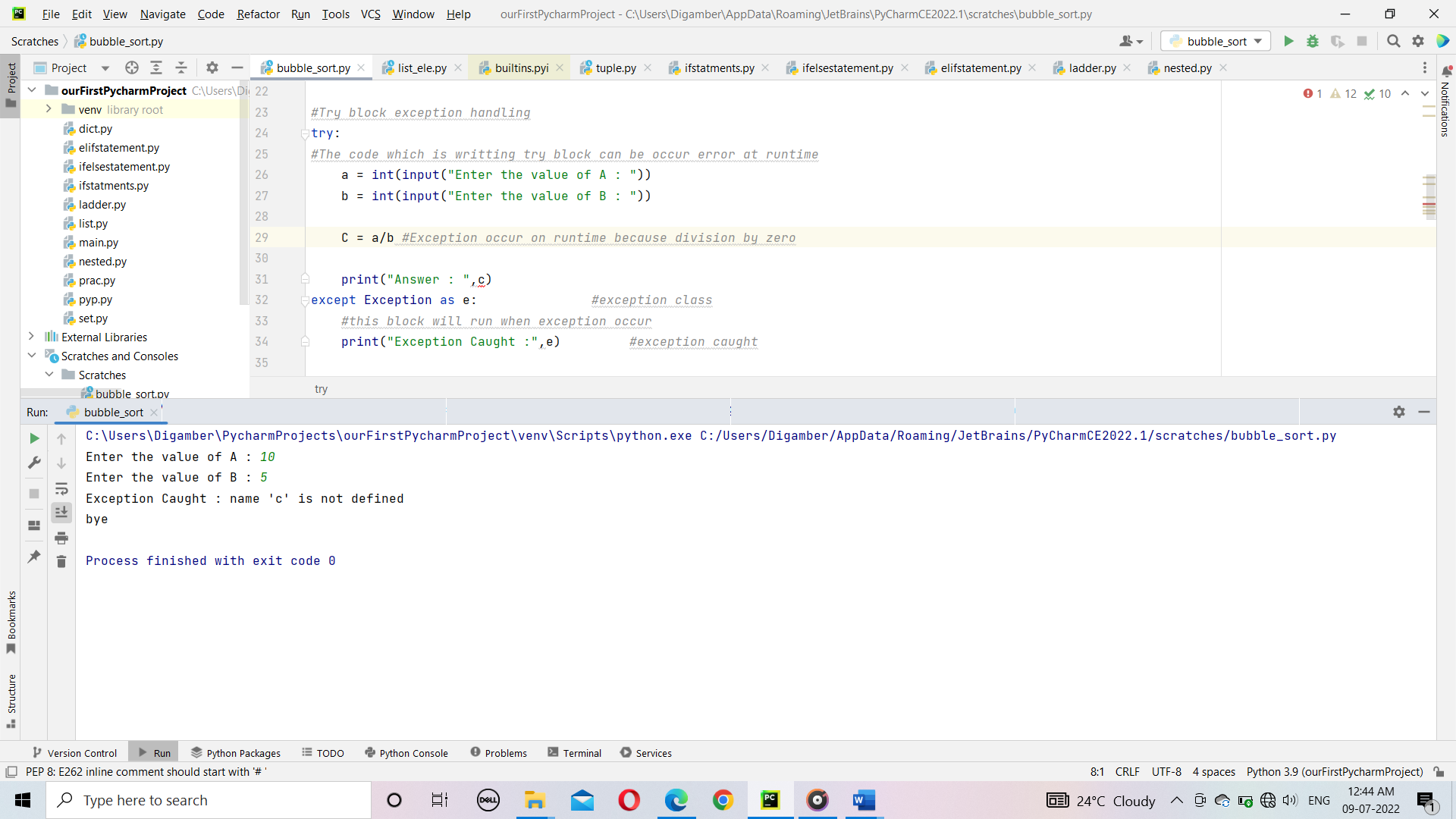


7. Write a programme to demonstrate the Exception handling in python.

try:  
*#The code which is writting try block can be occur error at runtime* a = int(input("Enter the value of A : "))  
 b = int(input("Enter the value of B : "))  
  
 C = a/b *#Exception occur on runtime because division by zero* print("Answer : ",c)  
except Exception as e: *#exception class  
 #this block will run when exception occur* print("Exception Caught :",e) *#exception caught*print("bye")

OUTPUT :





1. Write a python programme for use of Linear search algorithm to search element in list.

5. Develop programs to understand the control structures of python.

Input : arr[] = {10, 20 ,80,30,60,50,110,100,130,170}

X = 110;

Output : 6

i/p : arr[] = { 10,20,80,30,60,50,110,100,130,170}

x=175;

o/p : -1

Element x is not present in arr[].

# def search (arr, x):

For i in range(len(arr)):

If arr[i] == x:

return -1