Bahria University, Islamabad

Department of Software Engineering

Artificial Intelligence Lab

(Fall-2021)

Teacher: Engr. M Waleed Khan

Student : M Iqrar Ijaz Malik

Enrollment : 01-131182-021

Lab Journal: 10

Date: 10-11-2021

Task No:	Task Wise Marks		Documentation Marks		Total Marks (20)
	Assigned	Obtained	Assigned	Obtained	
1	15		5		

Comments:

Signature

LAB

Lab 10: Implementing Sorting Algorithms

Introduction

Sometimes, data we store or retrieve in an application can have little or no order. We may have to rearrange the data to correctly process it or efficiently use it. Over the years, computer scientists have created many sorting algorithms to organize data.

- Bubble Sort
- Selection Sort
- Insertion Sort
- Merge Sort
- Heap Sort
- Quick Sort
- Sorting in Python

Bubble Sort

This simple sorting algorithm iterates over a list, comparing elements in pairs and swapping them until the larger elements "bubble up" to the end of the list, and the smaller elements stay at the "bottom".

Time Complexity

In the worst case scenario (when the list is in reverse order), this algorithm would have to swap every single item of the array. Our swapped flag would be set to True on every iteration. Therefore, if we haven elements in our list, we will have n iterations per item - thus Bubble Sort's time complexity is O(n^2).

Selection Sort

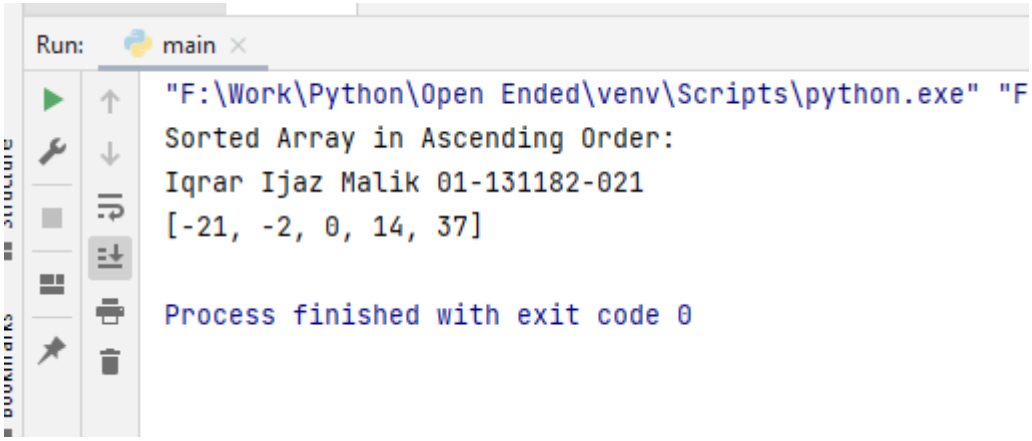
This algorithm segments the list into two parts: sorted and unsorted. We continuously remove the

smallest element of the unsorted segment of the list and append it to the sorted segment.

Task 1

```
# Bubble sort in Python
def bubbleSort(array):
 for i in range(len(array)):
 # loop to compare array elements
    for j in range(0, len(array) - i - 1):
         if array[j] > array[j + 1]:
             temp = array[j]
             array[j] = array[j + 1]
             array[j + 1] = temp
data = [-2, 37, 0, 14, -21]
bubbleSort(data)
print('Sorted Array in Ascending Order:')
print("Iqrar Ijaz Malik 01-131182-021")
print(data)
```

Output:



Conclusion

I completed the tasks given to us and pasted output above.