Aryan Sharma

BT21GCS161

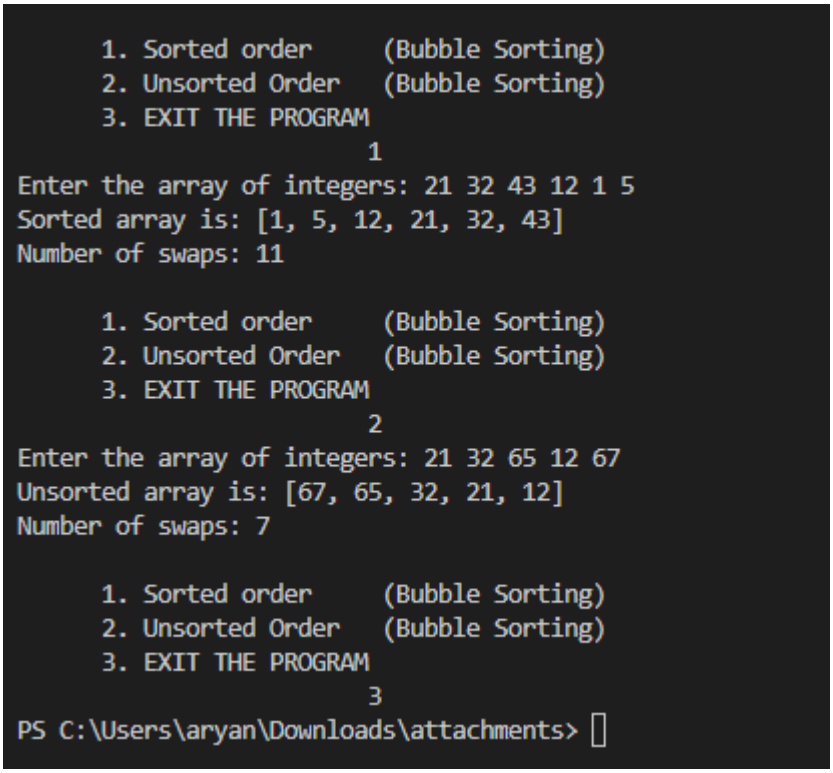
Assignment -04

Code

10.py > ...

```
choice=1
while(choice==1 or choice==2 or choice==3):
    choice=int(input('''
      1. Sorted order     (Bubble Sorting)
      2. Unsorted Order   (Bubble Sorting)
      3. EXIT THE PROGRAM
                        '''))

# Bubble Sorting (1.Sorted Order)
    if(choice ==1):
        a = [int(x) for x in input("Enter the array of integers: ").split()]
        n = len(a)
        S = 0 #swap
        for i in range(n):
           for j in range(0, n-i-1):
            if a[j] > a[j+1]:
                a[j], a[j+1] = a[j+1], a[j]
                S += 1

        print("Sorted array is:", a)
        print("Number of swaps:", S)

# Bubble Sorting (2. UnSorted Order)

    elif(choice ==2):
        b = [int(x) for x in input("Enter the array of integers: ").split()]
        n = len(b)
        S = 0 #Swap
        for i in range(n):
            for j in range(0, n-i-1):
                if b[j] < b[j+1]:
                    b[j], b[j+1] = b[j+1], b[j]
                    S += 1
        print("Unsorted array is:", b)
        print("Number of swaps:", S)

# Exit
    elif(choice ==3):
        break


```

Output-



Analysis-

Best Case-

The specified array is sorted by default when it is used, which is the best-case scenario.

Time Complexity is O(n)

Space Complexity is O(1)

Worst Case-

The array being sorted in the incorrect way, such as ascending when descending order is required, or vice versa is the worst-case scenario.

Time Complexity is O(n^2)

Space Complexity is O(1)

Average Case-

When the list is in random order, the average case occurs.

Time Complexity is O(n^2)

Space Complexity is O(1)