Sorting - Programs

Write python program for the following sorting algorithms.
1.Selection sort
2.Quick sort
3.Merge sort
4.Bubble sort
5.Insertion sort

1. Selection Sort

```python
#selection sort

arr = [13,12,9,7,11]
n = len(arr)

for i in range(n):
    min_index = i
    for j in range(i+1, n):
        if arr[j] < arr[min_index]:
            min_index = j
    arr[i], arr[min_index] = arr[min_index], arr[i]

print("Sorted array:", arr)
```

Output:

```
Sorted array: [7, 9, 11, 12, 13]
```

2. Quick Sort

```python
arr = [13,12,9,7,11]
def quick_sort(arr, low, high):
    if low < high:
        pivot_index = partition(arr, low, high)
        quick_sort(arr, low, pivot_index - 1)
        quick_sort(arr, pivot_index + 1, high)

def partition(arr, low, high):
    pivot = arr[high]
    i = low - 1
    for j in range(low, high):
        if arr[j] <= pivot:
            i += 1
            arr[i], arr[j] = arr[j], arr[i]
    arr[i + 1], arr[high] = arr[high], arr[i + 1]
    return i + 1

quick_sort(arr, 0, len(arr) - 1)
print("Sorted array:", arr)
```

Output:

```
Sorted array: [7, 9, 11, 12, 13]
```

3. Merge Sort

```python
arr = [13,12,9,7,11]
def merge_sort(arr):
    if len(arr) <= 1:
        return arr
    mid = len(arr) // 2
    left_half = arr[:mid]
    right_half = arr[mid:]
    left_sorted = merge_sort(left_half)
    right_sorted = merge_sort(right_half)
    return merge(left_sorted, right_sorted)

def merge(left, right):
    result = []
    i = j = 0
    while i < len(left) and j < len(right):
        if left[i] <= right[j]:
            result.append(left[i])
            i += 1
        else:
            result.append(right[j])
            j += 1
    result.extend(left[i:])
    result.extend(right[j:])
    return result

sorted_arr = merge_sort(arr)
print("Sorted array:", sorted_arr)
```

```
Sorted array: [7, 9, 11, 12, 13]
```

Output:

4. Bubble Sort

```python
#bubble sort

arr = [13,25,83,272,82,825,8]

n = len(arr)

for i in range(n-1):
 for j in range(n-i-1):
   if arr[j] > arr[j+1]:
     arr[j], arr[j+1] = arr[j+1], arr[j]

print("Sorted array:", arr)
```

Output:

```
Sorted array: [8, 13, 25, 82, 83, 272, 825]
```

5. Insertion Sort

```python
numbers = [5, 2, 9, 1, 5, 6]

for i in range(1, len(numbers)):
    key = numbers[i]
    j = i - 1
    while j >= 0 and key < numbers[j]:
        numbers[j + 1] = numbers[j]
        j -= 1
    numbers[j + 1] = key

print(numbers)
```

Output:

```
[1, 2, 5, 5, 6, 9]
```

Submitted by - Nanda Krishnan V , Msc Cybersecurity