Exp-2.2

Title:

Selection Sort Algorithm Explanation and Examples

Aim:

To explain and implement the Selection Sort algorithm to sort an array of integers in ascending order.

Procedure:

1. Divide the array into sorted and unsorted parts; initially, the sorted part is empty.
2. Repeatedly find the smallest element in the unsorted part.
3. Swap this smallest element with the first unsorted element.
4. Move the boundary of the sorted part one element to the right.
5. Repeat until all elements are sorted.

Algorithm:

1. Start with index i = 0.
2. For each position i in the array (from 0 to n-1):
3. Find the index min_idx of the smallest element in the subarray from i to end.
4. Swap elements at indices i and min_idx.
5. Increment i to move forward.
6. Repeat until the array is fully sorted.
7. End.

Input:

6

5 2 9 1 5 6

5

10 8 6 4 2

5

1 2 3 4 5

Output:

1 2 5 5 6 9

2 4 6 8 10

1 2 3 4 5

Program:

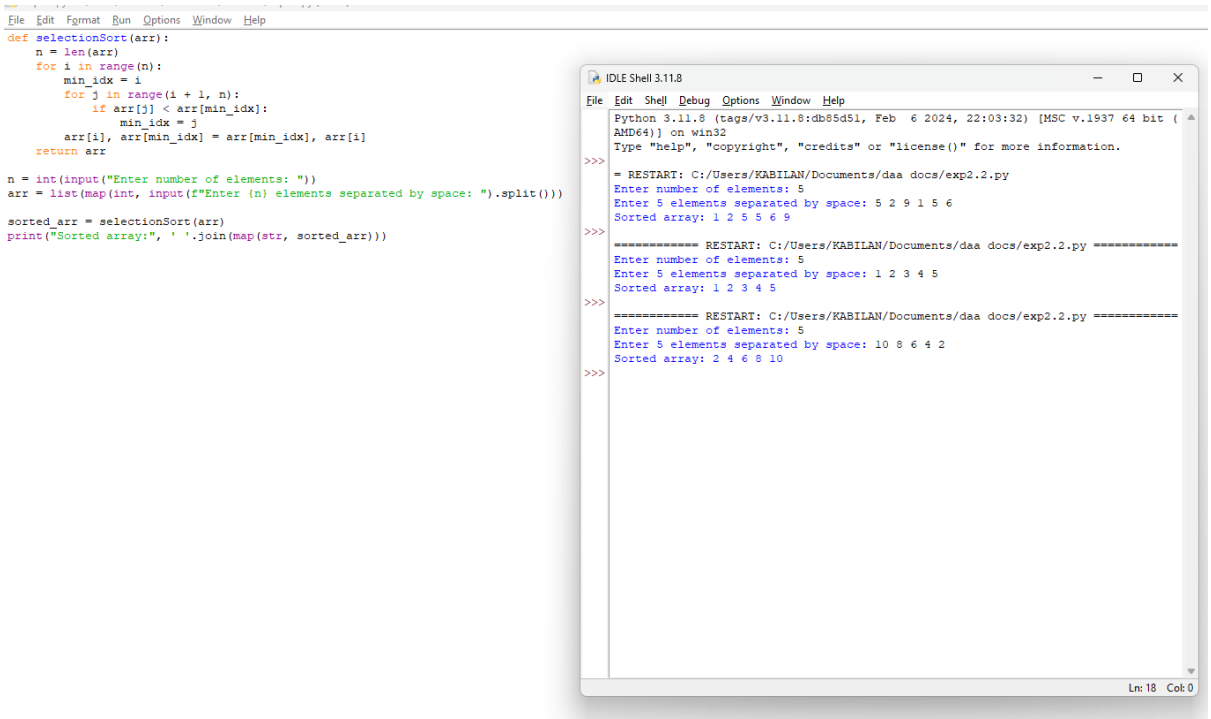
```
def selectionSort(arr):
    n = len(arr)
    for i in range(n):
        min_idx = i
        for j in range(i + 1, n):
            if arr[j] < arr[min_idx]:
                min_idx = j
        arr[i], arr[min_idx] = arr[min_idx], arr[i]
    return arr
n = int(input("Enter number of elements: "))
arr = list(map(int, input(f"Enter {n} elements separated by space: ").split()))

sorted_arr = selectionSort(arr)
print("Sorted array:", ' '.join(map(str, sorted_arr)))
```

Performance Analysis:

Time Complexity: $O(n^2)$.

Space Complexity: O(1)

Program Output:



Result:

Thus the given program Selection Sort is executed and got output successfully.