Name: Bharat Sharma
Sec & Year: K (3rd year)
Roll No. : 191599003

Task-3: Implementation of Selection Sort

Selection Sort: - Selection Sort is a simple sorting algorithm. This sorting algorithm is an in-place comparison-based algorithm in which the list is dividing into parts, the sorted part at the left end and the unsorted part at the right end. Initially, the sorted part is empty and the unsorted part is the entire list.

The smallest element is selected from the unsorted array and swapped with the leftmost element, and that element becomes a part of the sorted array. This process continues moving unsorted array boundary from one element to the right.

Java Program:

```
class SelectionSort
{
	public SelectionSort(){}
	public void selectionSort(int[] arr)
	{
		for(int i = 0; i < arr.length-1; i++)
		{
			int index = i;
			for(int j = i+1; j <arr.length; j++)
			{
				if(arr[j] < arr[index])
					index = j;
			}

			int smallerNumber = arr[index];
			arr[index] = arr[i];
			arr[i] = smallerNumber;
		}
	}
}

class SelectionSortDriver
{
	public static void main(String[] args)
	{
		int[] arr1 = {9,14,3,2,43,11,58,22};
		System.out.println("Before Selection Sort...");
```

```
        for(int i:arr1)
        {
            System.out.print(i +" ");
        }

        SelectionSort obj = new SelectionSort();
        obj.selectionSort(arr1);

        System.out.println("\nAfter Selection Sort...");
        for(int i : arr1)
        {
            System.out.print(i +" ");
        }
    }
}
```

Complexity Analysis

1. Worst Case Time Complexity [Big-O]: **$O(n^2)$**
2. Best Case Time Complexity [Big-omega]: **$O(n^2)$**
3. Average Time Complexity [Big-theta]: **$O(n^2)$**
4. Space Complexity: **O(1)**