Page No.
Date

Assignment No. 5

Name - Sumit Gulab Bhamare
Div - A Roll no - 8
Sub - Data Structure Lab

Aim - To illustrate the various sorting techniques.

Problem Statement: Write a Python program to store first year percentage of students in array. Write function for sorting array of floating-point numbers in ascending order using a) selection sort b) Bubble sort & display top five scores.

Learning Objectives: To understand concept of sorting
To compare time complexity of various sorting techniques.

Learning Outcome: Students will be able to analyze problems to apply suitable searching & sorting algorithm to various applications.

Theory:

Bubble Sort:
Bubble sort is a simple & well-known sorting algorithm. It is used in practice once in a blue moon & its main application is to make an introduction to the sorting algorithms. Bubble sort belongs to $O(n^2)$ sorting algorithms, which makes it quite inefficient for sorting large data volumes. Bubble sort is stable & adaptive.

5 1 12 -5 16	unsorted
[5] [1] 12 -5 16	5>1, swap
1 [5] [12] -5 16	5<12, ok
1 5 [12] [-5] 16	12>-5, swap
1 5 -5 [12] [16]	12<16, ok
[1] [5] -5 12 16	1<5, ok
1 [5] [-5] 12 16	1>-5, swap
1 -5 [5] [12] 16	1<5, ok
[1] [-5] 5 12 16	-5<1, ok
-5 1 5 12 16	sorted

Selection sort:-

Selection sort is one of the $O(n^2)$ sorting algorithms, which makes it quite inefficient for sorting large data volumes. Selection sort is notable for its programming simplicity & it can over perform other sorts in certain situations (see complexity analysis for more details).

5 1 12 -5 16 2 12 14

5 1 12 -5 16 2 12 14

-5 1 12 5 16 2 12 14

-5 1 12 5 16 2 12 14

-5 1 2 5 16 12 12 14

-5 1 2 5 16 12 12 14

-5 1 2 5 12 16 12 14

-5 1 2 5 12 12 16 14

-5 1 2 5 12 12 14 16

Input : Enter the first percentage of students.

Output : Sorting by selection & bubble sort & display top 5 student marks.

Algorithm for Bubble sort :-
We assume list is an array of n elements, We further assume the swap function, swaps the values of given array elements

```
begin BubbleSort(list)
  for all elements of list
    if list[i] > list[i+1]
      swap(list[i], list[i+1])
    end if
  end for
  return list
end BubbleSort.
```

Time Complexity = O(n^2)

Algorithm for Selection Sort :
We assume list is an array of n elements. We further assume that swap function, swaps the values of given array elements.

```
Begin SelectionSort(list)
    for all elements of list
        for j to all element of list
            if list[i] > list[j]
                swap(list[i], list[j])
            end if
        end for
    end for
    return list.
end BubbleSort.
```

Software required: Open source Python, Programming tool like Jupyter Notebook, Pycharm, Spyder.

Conclusion: Thus, we have studied the implementation of various sorting techniques (Bubble & Selection)