# Experiment No- 06

**Title:-** Python program to compute Insertion Sort and Shell sort algorithm

**Objectives:-** To understand the use of sorting using algorithm

# Problem Statement:-

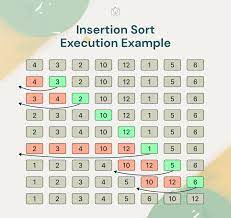
Write a Python program to store second year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using

1. Insertion sort
2. Shell Sort and display top five scores

# Theory:-

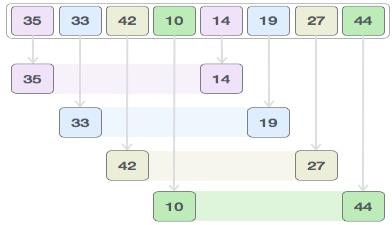
**Insertion sort algorithm**

1. Consider the first element to be sorted and the rest to be unsorted.
2. Take the first element in the unsorted part(u1) & compare it with sorted part elements(s1).
3. if u1<s1 then insert ul in the correct index, else leave it as it is.
4. Take next element in the unsorted part and
5. Repeat 3 and 4 until all the elements are sorted.

****

**Shell Sort algorithm:-**

1. Take the list of numbers
2. Find out the gap/incrementor
3. Create the sublist based on gap and sort them using insertion sort algorithm.
4. Reduce gap and repeat step 3.
5. Stop when gap is 0. """

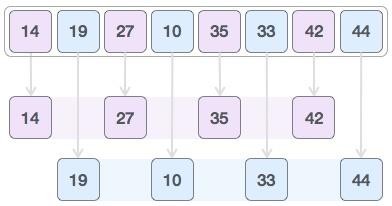
Consider the following array: 35,33,42,10,14,19,27,44

We compare values in each sub-list and swap them (if necessary) in the original array. After this step, the new array should look like this −



Then, we take interval of 1 and this gap generates two sub-lists - {14, 27, 35, 42}, {19, 10,

33, 44}



We compare and swap the values, if required, in the original array. After this step, the array should look like this −



Finally, we sort the rest of the array using interval of value 1. Shell sort uses insertion sort to sort the array.

Following is the step-by-step depiction −



We see that it required only four swaps to sort the rest of the array.

# Conclusion:

In this way, we perform Sorting of marks using insertion sort and shell sort algorithm.