Experiment No. 1

Aim: Implement Insertion Sort

Theoretical Background:

Insertion sort is a sorting algorithm that places an unsorted element at its suitable place in each iteration. Insertion sort works similarly as we sort cards in our hand in a card game.

- **Algorithm:**

```
for j = 2 to A.length
    key = A[j]
    i = j – 1
    while i > 0 and A[i] > key
        A[i + 1] = A[i]
        i = i -1
    A[i + 1] = key
```

Program:

```
    #include <math.h>
    #include <stdio.h>
     #include<conio.h>
/* Function to sort an array using insertion sort*/
void insertionSort(int arr[], int n)
{
    int i, key, j;
    for (i = 1; i < n; i++) {
        key = arr[i];
        j = i - 1;
        /* Move elements of arr[0..i-1], that are
          greater than key, to one position ahead
          of their current position */
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;
    }
}
// A utility function to print an array of size n
void printArray(int arr[], int n)
{
    int i;
    printf("\n Sorted array:");
    for (i = 0; i < n; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
/* Driver program to test insertion sort */
int main()
{
    int arr[50],i,n;

    clrscr();

    printf("\nEnter the size of array:") ;

    scanf("%d",&n);

    printf("\nEnter array elements:");

    for(i=0;i<n;i++)

    {

    scanf("%d",&arr[i]);

    }

    insertionSort(arr, n);

    printArray(arr, n);

    getch();

    return 0;

}
```

Output:

Conclusion:

Thus, in this experiment we have studied about Insertion Sort.