Program 11

Ques 11) WAP to Sort a List of Numbers Using Insertion Sort?

Sol:

Algorithm:

1) Start.
2) Initialize:
   a) Declare an integer array arr[5].
   b) Declare an integer variable n to store the size of the array.
3) Input the size of the array:
   a) Print "Enter the size of an array:".
   b) Take the user input and store it in n.
4) Input the elements of the array:
   a) Print "Enter the elements of an array:".
   b) Loop from i = 0 to n-1:
      i) Take the user input for each element and store it in arr[i].
5) Sort the array using Insertion Sort:
   a) Loop from i = 1 to n-1:
      i) Set curr = arr[i] (store the current element).
      ii) Set prev = i - 1 (index of the previous element).
      iii) Inner loop to find the correct position:
         (1) While prev >= 0 and arr[prev] > curr:
            (a) Shift arr[prev] to the right by setting arr[prev + 1] = arr[prev].
            (b) Decrement prev by 1.
      iv) Insert the current element:
         (1) Set arr[prev + 1] = curr.
6) Print the sorted array:
   a) Loop from i = 0 to n-1:
      i) Print arr[i] followed by a space.
7) End.

Code:

```
#include<stdio.h>

int main(){

    int arr[5];
```

```c
    int n;
    printf("Enter the Size of an array:");
    scanf("%d",&n);
    printf("Enter the Elements of an Array:");
    for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }

    for (int i = 1; i < n; i++)  {//no. of turns
        int curr = arr[i];
        int prev = i - 1;
        // finding the position to insert
        while (prev >= 0 && arr[prev] > curr) {
            arr[prev + 1] = arr[prev];
            prev--;
        }
        // insertion
        arr[prev + 1] = curr;
    }

    for(int i=0;i<n;i++){
        printf("%d ",arr[i]);
    }
    return 0;
}
```

Output:

```
Enter the Size of an array:5
Enter the Elements of an Array:5
4
3
2
1
1 2 3 4 5
```