Insertion sort---→>>

```c
#include <math.h>

#include <stdio.h>



void insertionSort(int arr[], int n)

{

        int i, key, j;

        for (i = 1; i < n; i++) {

                key = arr[i];

                j = i - 1;


        while (j >= 0 && arr[j] > key) {

                        arr[j + 1] = arr[j];

                        j = j - 1;

                }

                arr[j + 1] = key;

        }

}



void printArray(int arr[], int n)

{

        int i;

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

                printf("%d ", arr[i]);

        printf("\n");

}
```

```c
int main()
{
        int arr[] = { 12, 11, 13, 5, 6 };
        int n = sizeof(arr) / sizeof(arr[0]);


        insertionSort(arr, n);
        printArray(arr, n);


        return 0;
}
```

Mergesort--- →>>

```c
#include<stdio.h>
void merge(int a[], int mid , int low , int high){
    int i,j,k;int b[100];
    i=low;
    j=mid+1;
    k=low;
    while(i<=mid && j<=high){
        if(a[i]<a[j]){
            b[k]=a[i];
            i++; k++;
        }
        else{
            b[k]=a[j];
            j++; k++;
```

```c
      }

    }

    while (i<=mid)

    {

       b[k]=a[i];

         i++; k++;

    }

     while (j<=high)

    {

       b[k]=a[j];

         j++; k++;

    }

    for (int i = low; i <=high; i++)

    {

       a[i]=b[i];

    }



}

void printArray(int a[],int n){

    for (int i = 0; i <=7; i++)

    {

       printf("%d ",a[i]);

    }

    printf("\n");

}

void mergeShort(int a[], int low, int high){

    int mid;

    if(low<high){
```

```c
        mid=(high+low)/2;

        mergeShort(a, low, mid);

        mergeShort(a,mid+1, high);

        merge(a, mid, low, high);

        }


}
int main(){

    int a[]={12,2,76,5,87,23,90};

    int n=7;

    printArray(a,n);

    mergeShort(a,0,6);

    printArray(a,n);

}
```