```cpp
#include<iostream>
using namespace std;

/*--------Merge Sort( O(nlogn) for all cases )--------*/

void inputArray(int arr[], int n)
{
    cout << "Enter the elements of your array: \n";
    for(int i = 0; i < n; i++)
    {
        cin >> arr[i];
    }
}

void printArray(int arr[], int n)
{
    for(int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
}

void swap(int arr[], int i, int j)
{
    int temp = arr[i];
    arr[i] = arr[j];
    arr[j] = temp;
}

void merge(int arr[], int left, int mid, int right)
{
    int i = left;
    int j = mid + 1;
    int k = 0;
    int temparr[right - left + 1];

    while(i <= mid && j <= right)
    {
        if(arr[i] <= arr[j])
        {
            temparr[k] = arr[i];
            k++;
            i++;
        }
        else
        {
```

```
47:             temparr[k] = arr[j];
48:             k++;
49:             j++;
50:         }
51:     }
52:     if(i > mid)
53:     {
54:         while(j <= right)
55:         {
56:             temparr[k] = arr[j];
57:             j++;
58:             k++;
59:         }
60:     }
61:     else
62:     {
63:         while(i <= mid)
64:         {
65:             temparr[k] = arr[i];
66:             i++;
67:             k++;
68:         }
69:     }
70:
71:     for(i = left; i <= right; i++)
72:     {
73:         arr[i] = temparr[i - left];
74:     }
75: }
76:
77: void mergeSort(int arr[], int left, int right)
78: {
79:     if(left < right)
80:     {
81:         int mid = (right + left) / 2;
82:
83:         mergeSort(arr, left, mid);
84:         mergeSort(arr, mid + 1, right);
85:
86:         merge(arr, left, mid, right);
87:     }
88: }
89:
90: int main()
91: {
92:     int n;
```

```
 93:     cout << "Enter the length: \n";
 94:     cin >> n;
 95:
 96:     int arr[n];
 97:     inputArray(arr, n);
 98:
 99:     mergeSort(arr, 0, n - 1);
100:
101:     cout << "Sorted array is: \n";
102:     printArray(arr, n);
103:
104:     return 0;
105: }
```