Lab Assignment – 3

WAP for implementing Merge Sort

Merge Sort Code:

```
package rec_Program;


import java.util.Scanner;
import java.util.Arrays;
public class MergeSort {


        void merge(int arr[], int p, int q, int r) {


         int n1 = q - p + 1;
         int n2 = r - q;


         int left[] = new int[n1];
         int right[] = new int[n2];


         for (int i = 0; i < n1; i++)
           left[i] = arr[p + i];
         for (int j = 0; j < n2; j++)
           right[j] = arr[q + 1 + j];


         int i, j, k;
         i = 0;
         j = 0;
```

```
k = p;

while (i < n1 && j < n2) {
  if (left[i] <= right[j]) {
    arr[k] = left[i];
    i++;
  }
  else
  {
    arr[k] = right[j];
    j++;
  }
  k++;
}

// When we run out of elements in either L or M,
// pick up the remaining elements and put in A[p..r]
while (i < n1) {
  arr[k] = left[i];
  i++;
  k++;
}

while (j < n2) {
  arr[k] = right[j];
  j++;
  k++;
```

```
 }
}

void mergeSort(int array[], int left, int right) {
        if (left < right)
        {
          int mid = (left + right) / 2;
          mergeSort(array, left, mid);
          mergeSort(array, mid + 1, right);
          merge(array, left, mid, right);
        }
       }

public static void main(String[] args) {
       // TODO Auto-generated method stub

       Scanner sc = new Scanner(System.in);
       System.out.println("Enter the Size of Array");
       int n = sc.nextInt();
       int[] arr = new int[n];

       System.out.println("Enter the elements of Array");
       for(int i =0;i<n;i++)
       {
              arr[i] = sc.nextInt();
       }
```

```
        MergeSort ob = new MergeSort();

        ob.mergeSort(arr, 0, n - 1);

        System.out.println("Sorted Array:");
        System.out.println(Arrays.toString(arr));
    }
}
```

Merge Sort Code output: