**package** Sorting\_Algorithms;

**public class** MergeSortAlgorithm {

**public int**[] **tempArray**;

**public int**[] **inputArray**;

**public int**[] **arra**;

**public void** mergeSort(**int**[] inputArray)

{

**this**.**tempArray** = inputArray;

**this**.**inputArray** = **new int**[**tempArray**.**length**];

partition(0,**tempArray**.**length**-1);

}

**public void** partition(**int** lowerIndex, **int** higherIndex)

{

**if** (lowerIndex<higherIndex)

{

**int** mid = lowerIndex + (higherIndex-lowerIndex)/2;

partition(lowerIndex,mid);

partition(mid+1,higherIndex);

doMerge(lowerIndex,mid,higherIndex);

}

}

**public void** doMerge(**int** lowerIndex,**int** middlePoint, **int** higherIndex)

{

**for** (**int** i=0;i<**tempArray**.**length**;i++)

{

**inputArray**[i] = **tempArray**[i];

}

**int** i = lowerIndex;

**int** j = middlePoint + 1;

**int** k = lowerIndex;

**while** (i<=middlePoint && j<=higherIndex)

{

**if** (**inputArray**[i]<=**inputArray**[j])

{

**tempArray**[k] = **inputArray**[i];

i++;

}**else**

{

**tempArray**[k] = **inputArray**[j];

j++;

}

k++;

}

**if** (j>higherIndex)

{

**while** (i<=middlePoint)

{

**tempArray**[k] = **inputArray**[i];

k++;

i++;

}

}

**if** (i>middlePoint)

{

**while** (j<=higherIndex)

{

**tempArray**[k] = **inputArray**[j];

j++;

k++;

}

}

}

**public static void** main(String[] args) {

**int**[] inputArray = {4,10,1,2,50,3,-1,-6,99};

MergeSortAlgorithm object = **new** MergeSortAlgorithm();

object.mergeSort(inputArray);

**for** (**int** i : object.**tempArray**)

{

System.***out***.print(i+**" "**);

}

}

}