//Assignment no 12

import java.util.Scanner;

class MergeSort

{

// Merge two subarrays L and M into arr

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

{

int n1 = q - p + 1;

int n2 = r - q;

int L[] = new int[n1];

int M[] = new int[n2];

for (int i = 0; i < n1; i++)

L[i] = arr[p + i];

for (int j = 0; j < n2; j++)

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

// Maintain current index of sub-arrays and main array

int i, j, k;

i = 0;

j = 0;

k = p;

// Until we reach either end of either L or M, pick larger among

// elements L and M and place them in the correct position at A[p..r]

while (i < n1 && j < n2)

{

if (L[i] <= M[j])

{

arr[k] = L[i];

i++;

} else

{

arr[k] = M[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] = L[i];

i++;

k++;

}

while (j < n2)

{

arr[k] = M[j];

j++;

k++;

}

}

// Divide the array into two subarrays, sort them and merge them

void mergeSort(int arr[], int l, int r)

{

if (l < r)

{

// m is the point where the array is divided into two subarrays

int m = (l + r) / 2;

mergeSort(arr, l, m);

mergeSort(arr, m + 1, r);

// Merge the sorted subarrays

merge(arr, l, m, r);

}

}

// Print the array

static void printArray(int arr[])

{

int n = arr.length;

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

System.out.print(arr[i] + " ");

System.out.println();

}

// Driver program

public static void main(String args[]) {

int n;

Scanner sc=new Scanner(System.in);

System.out.print("Enter the number of elements you want to store: ");

//reading the number of elements from the that we want to enter

n=sc.nextInt();

int[] arr = new int[n];

System.out.println("Enter the elements of the array: ");

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

{

//reading array elements from the user

arr[i]=sc.nextInt();

}

System.out.println("Enter Array before sorting");

System.out.println("Array before sorting");

printArray(arr);

MergeSort ob = new MergeSort();

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

System.out.println("Sorted array:");

printArray(arr);

}

}

output:

gescoe@gescoe-OptiPlex-3010:~/Desktop/SE-A-55$ javac MergeSort.java

gescoe@gescoe-OptiPlex-3010:~/Desktop/SE-A-55$ java MergeSort

Enter the number of elements you want to store: 5

Enter the elements of the array:

6

3

7

2

5

Enter Array before sorting

Array before sorting

6 3 7 2 5

Sorted array:

2 3 5 6 7

gescoe@gescoe-OptiPlex-3010:~/Desktop/SE-A-55$