**public** **class** DivideandConquer {

**private** **static** **int**[] *inputsequence* = {4,2,1,3,7,6,5,8};

**public** **void** mergesort(**int**[] inputsequence, **int** p, **int** r)

{

**if**(p<r)

{

**int** q = (p+r)/2;

// System.out.println("Q is "+q);

mergesort(inputsequence, p, q);

mergesort(inputsequence, q+1,r);

merge1(inputsequence,p,q,r);

}

}

**public** **void** merge1(**int**[] inputsequence,**int** p, **int** q, **int** r)

{

**int** n1 = q-p+1;

**int** n2 = r-q;

**int** i,j;

**int**[] L1 = **new** **int**[n1];

**int**[] R1 = **new** **int**[n2];

**for**(**int** k=0;k<n1;k++)

{

L1[k]= inputsequence[p+k];

}

**for**(**int** m=0;m<n2;m++)

{

R1[m]= inputsequence[q+m+1];

}

// L1[n1+1]= 100;

// R1[n2+1]= 100;

i=0;

j=0;

**int** s = p;

**while** (i < n1 && j < n2)

{

**if**(L1[i]<=R1[j])

{

inputsequence[s]=L1[i];

i++;

s++;

}

**else**

{

inputsequence[s]=R1[j];

j++;

s++;

}

}

}

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

{

// Find sizes of two subarrays to be merged

**int** n1 = q - p + 1;

**int** n2 = r - q;

/\* Create temp arrays \*/

**int** L[] = **new** **int** [n1];

**int** R[] = **new** **int** [n2];

/\*Copy data to temp arrays\*/

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

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

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

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

/\* Merge the temp arrays \*/

// Initial indexes of first and second subarrays

**int** i = 0, j = 0;

// Initial index of merged subarry array

**int** k = p;

**while** (i < n1 && j < n2)

{

**if** (L[i] <= R[j])

{

arr[k] = L[i];

i++;

}

**else**

{

arr[k] = R[j];

j++;

}

k++;

}

/\* Copy remaining elements of L[] if any \*/

**while** (i < n1)

{

arr[k] = L[i];

i++;

k++;

}

/\* Copy remaining elements of R[] if any \*/

**while** (j < n2)

{

arr[k] = R[j];

j++;

k++;

}

}

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

// **TODO** Auto-generated method stub

**int** p,q,r;

p = 0;

r = *inputsequence*.length-1;

// q = (r-p)/2;

// System.out.println("p is : "+p+" q is : "+q+" r is :"+r);

DivideandConquer dc = **new** DivideandConquer();

dc.mergesort(*inputsequence*,0,*inputsequence*.length-1);

**for**(**int** h=0;h<*inputsequence*.length;h++)

System.***out***.println(*inputsequence*[h]);

}

}