Q: - Implement merge sort algorithm to merge two integer arrays into a third array in sorted order.

Input:-

static void Main(string[] args)

{

int[] A = new int[] { 5,7,11,13 };

int[] B = new int[] { 1, 2, 6, 8, 9 };

Console.WriteLine("Array A");

foreach (int arrA in A)

{

Console.Write(arrA+" ");

}

Console.WriteLine("\n\nArray B");

foreach (int arrB in B)

{

Console.Write(arrB+" ");

}

Console.WriteLine();

int[] C = new int[A.Length + B.Length];

int i, j, k;

for (i = 0, j = 0, k = 0; i < A.Length && j < B.Length; k++ )

{

if (A[i]<=B[j])

{

C[k] = A[i];

i++;

}

else

{

C[k] = B[j];

j++;

}

}

for (; i < A.Length;i++,k++ )

{

C[k] = A[i];

}

for (; j < B.Length;j++,k++ )

{

C[k] = B[j];

}

Console.WriteLine("\nAfter Merging A and B");

foreach (int item in C)

{

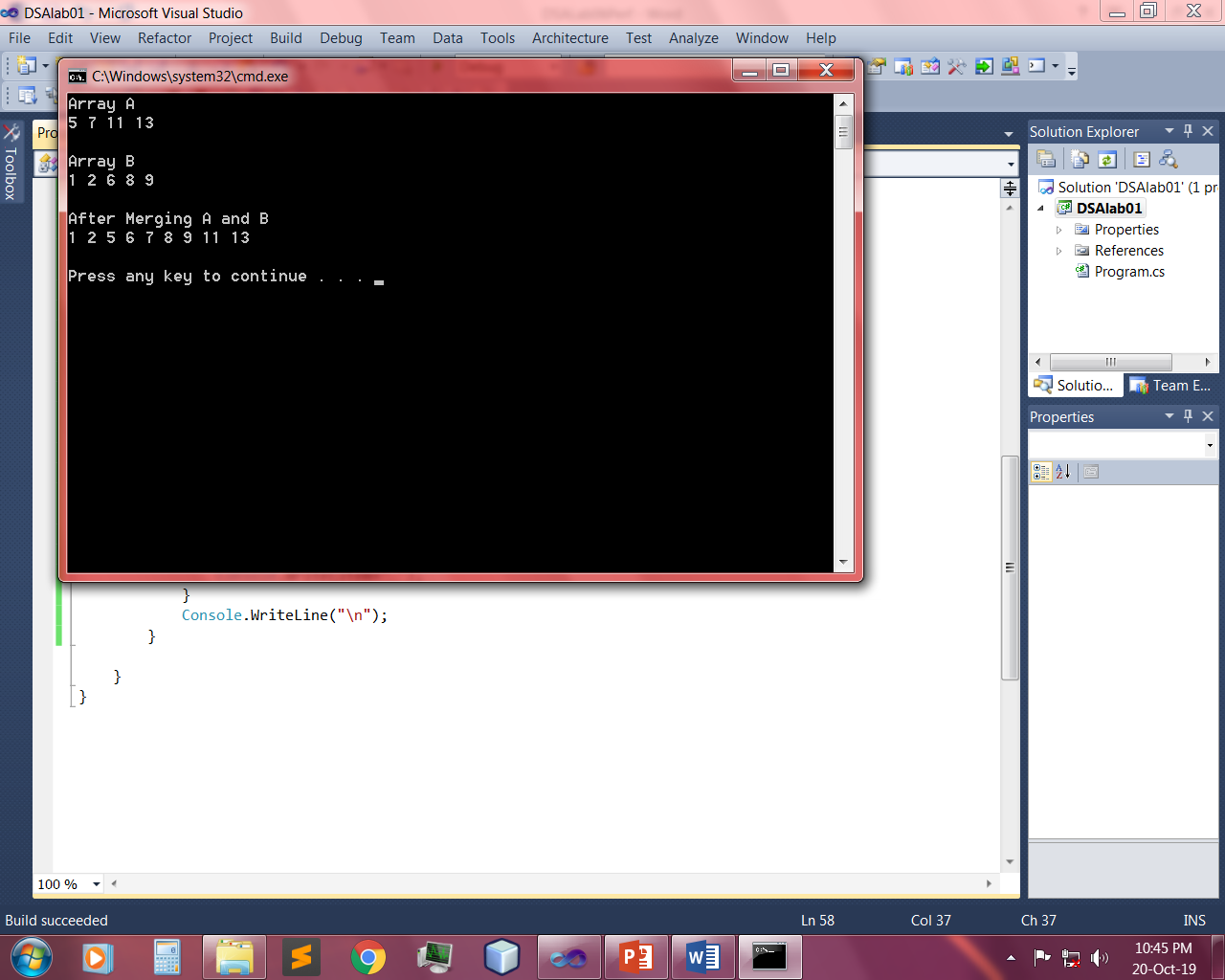
Console.Write(item+" ");

}

Console.WriteLine("\n");

}

Output:-



Q: - Implement recursive method of merge sort algorithm to sort an array of 10 characters.

Input:-

public static void MergeSort(char[] A,int lb,int ub)

{

int mid = (lb + ub) / 2;

if (lb<ub)

{

MergeSort(A, lb, mid);

MergeSort(A, mid + 1, ub);

Merge(A, lb, mid, mid + 1, ub);

}

}

public static void Merge(char[] A, int lb1, int ub1, int lb2, int ub2)

{

int i, j, k;

char[] C = new char[A.Length];

for (i = 0; i < A.Length; i++)

{

C[i] = A[i];

}

for (i = lb1,j=lb2,k=lb1; i <= ub1&&j<=ub2&&k<=ub2; k++)

{

if (C[i]<=C[j])

{

A[k] = C[i];

i++;

}

else

{

A[k] = C[j];

j++;

}

}

for (; i<=ub1; i++,k++)

{

A[k] = C[i];

}

for (; j <= ub2;k++,j++ )

{

A[k] = C[j];

}

}

static void Main(string[] args)

{

char[] A = new char[] { 'x','z','b','i','a','w','q' };

Console.WriteLine("Given Unsorted Array");

foreach (char item in A)

{

Console.Write(item + " ");

}

Console.WriteLine();

int lb = 0;

int ub = A.Length - 1;

MergeSort(A, lb, ub);

Console.WriteLine("\nSorted Array");

foreach (char item in A)

{

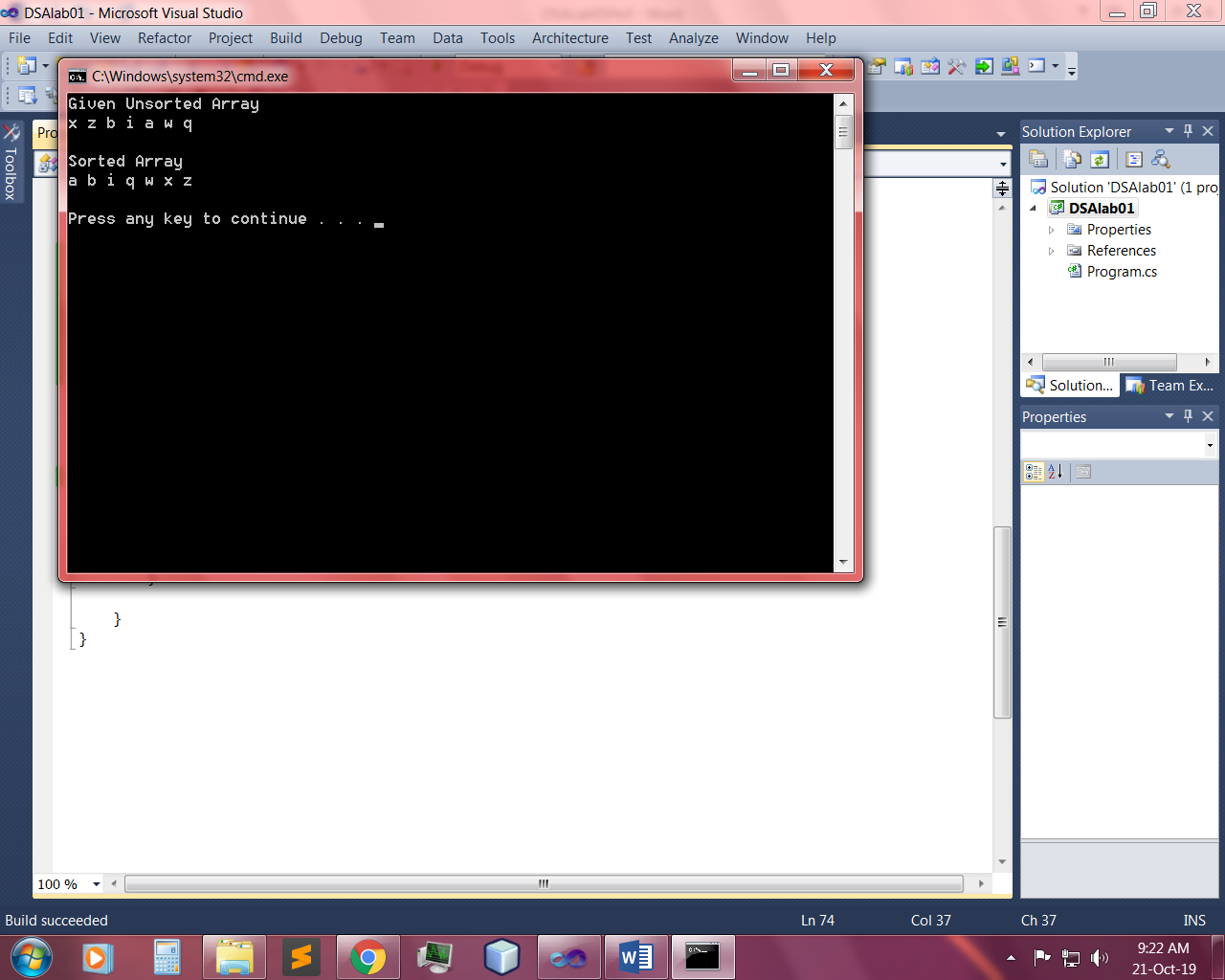
Console.Write(item+" ");

}

Console.WriteLine("\n");

}

Output:-



~~~~~~\*\*/**THE END**/\*\*~~~~~~

Q: - Implement recursive method of merge sort algorithm to sort an integer array.

Input:-

public static void MergeSort(int[] A,int lb,int ub)

{

int mid = (lb + ub) / 2;

if (lb<ub)

{

MergeSort(A, lb, mid);

MergeSort(A, mid + 1, ub);

Merge(A, lb, mid, mid + 1, ub);

}

}

public static void Merge(int[] A, int lb1, int ub1, int lb2, int ub2)

{

int i, j, k;

int[] C = new int[A.Length];

for (i = 0; i < A.Length; i++)

{

C[i] = A[i];

}

for (i = lb1,j=lb2,k=lb1; i <= ub1&&j<=ub2&&k<=ub2; k++)

{

if (C[i]<=C[j])

{

A[k] = C[i];

i++;

}

else

{

A[k] = C[j];

j++;

}

}

for (; i<=ub1; i++,k++)

{

A[k] = C[i];

}

for (; j <= ub2;k++,j++ )

{

A[k] = C[j];

}

}

static void Main(string[] args)

{

int[] A = new int[] { 5, 7, 2, 9, 3, 4, 1 };

Console.WriteLine("Given Unsorted Array");

foreach (int item in A)

{

Console.Write(item + " ");

}

Console.WriteLine();

int lb = 0;

int ub = A.Length - 1;

MergeSort(A, lb, ub);

Console.WriteLine("\nSorted Array");

foreach (int item in A)

{

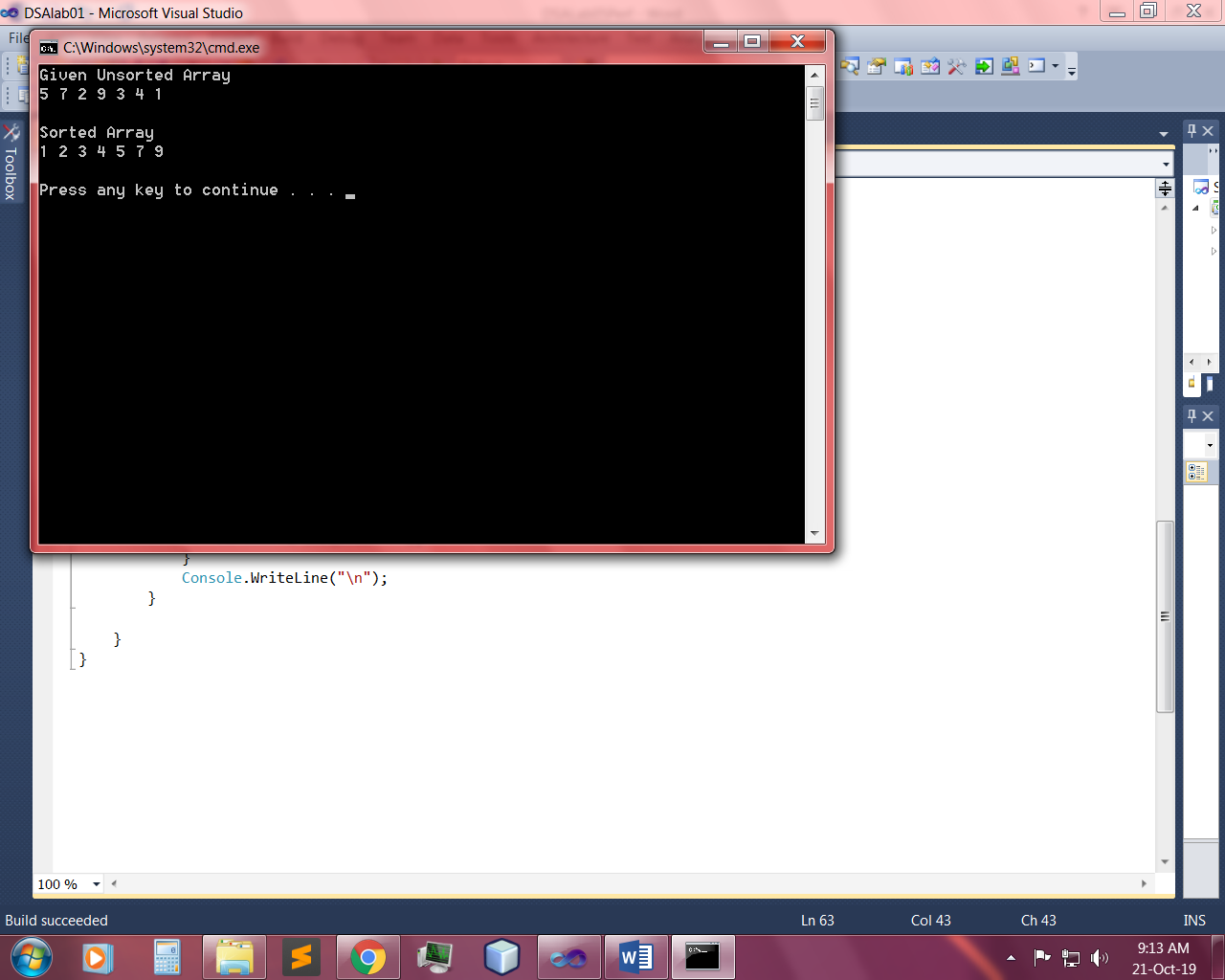
Console.Write(item+" ");

}

Console.WriteLine("\n");

}

Output:-



~~~~~~\*\*/**THE END**/\*\*~~~~~~