1. **To implement Quick sort.**

Method 1:

#include<stdio.h>

void quicksort(int ar[],int start, int end)

{

if(start>end)

return;

int i=start-1;

int pindex=end;

for(int j=start;j<end;j++)

{

if(ar[j]<ar[pindex])

{

i=i+1;

int temp=ar[j];

ar[j]=ar[i];

ar[i]=temp;

}

}

i=i+1;

int flag=ar[i];

ar[i]=ar[end];

ar[end]=flag;

quicksort(ar,start,i-1);

quicksort(ar,i+1,end);

}

void print(int ar[],int n)

{

printf("\nSorted elements are\n");

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

printf("%d ",ar[i]);

}

main()

{

int n;

printf("\nEnter the no of elements for sorting ");

scanf("%d",&n);

int ar[n];

printf("\nEnter the elements for sorting\n");

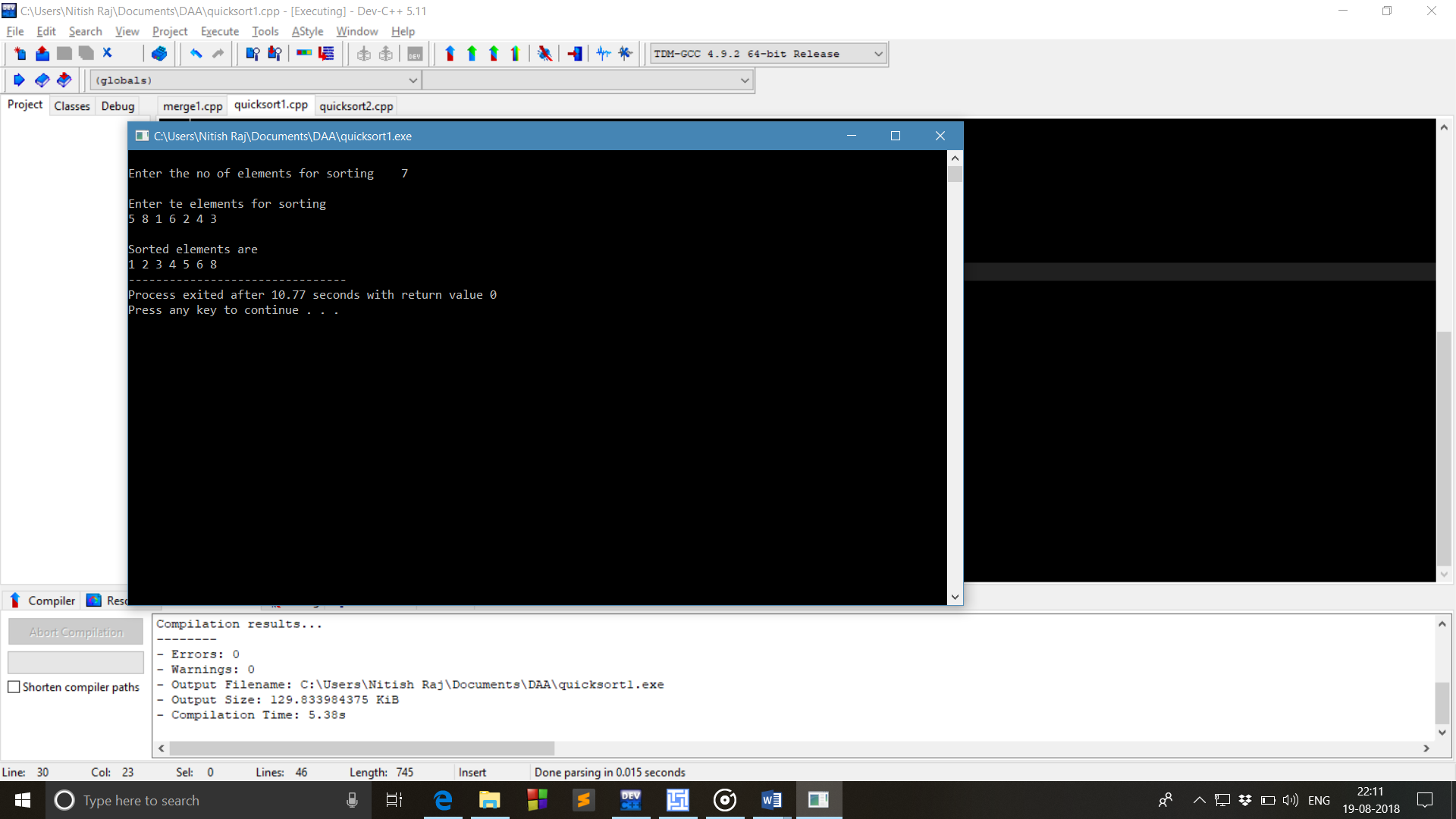
for(int i=0;i<n;i++)

scanf("%d",&ar[i]);

quicksort(ar,0,n-1);

print(ar,n);

}



Method 2:

#include<stdio.h>

int partition(int ar[],int low,int high)

{

int i,j,pivot,temp;

if(low<high)

{

pivot=low;

i=low;

j=high;

while(i<j)

{

while(ar[i]<ar[pivot]&&i<high)

i++;

while(ar[j]>ar[pivot])

j--;

if(i<j)

{

temp=ar[i];

ar[i]=ar[j];

ar[j]=temp;

}

}

temp=ar[pivot];

ar[pivot]=ar[j];

ar[j]=temp;

return j;

}

}

quicksort(int ar[],int low,int high)

{

int pivotloc;

if(low>high)

return 0;

pivotloc=partition(ar,low,high);

quicksort(ar,low,pivotloc-1);

quicksort(ar,pivotloc+1,high);

}

void print(int ar[],int n)

{

printf("\nSorted elements are\n");

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

printf("%d ",ar[i]);

}

int main()

{

int i,n;

printf("How much numbers you want to sort ");

scanf("%d",&n);

int ar[n];

printf("\nEnter the numbers for sorting\n");

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

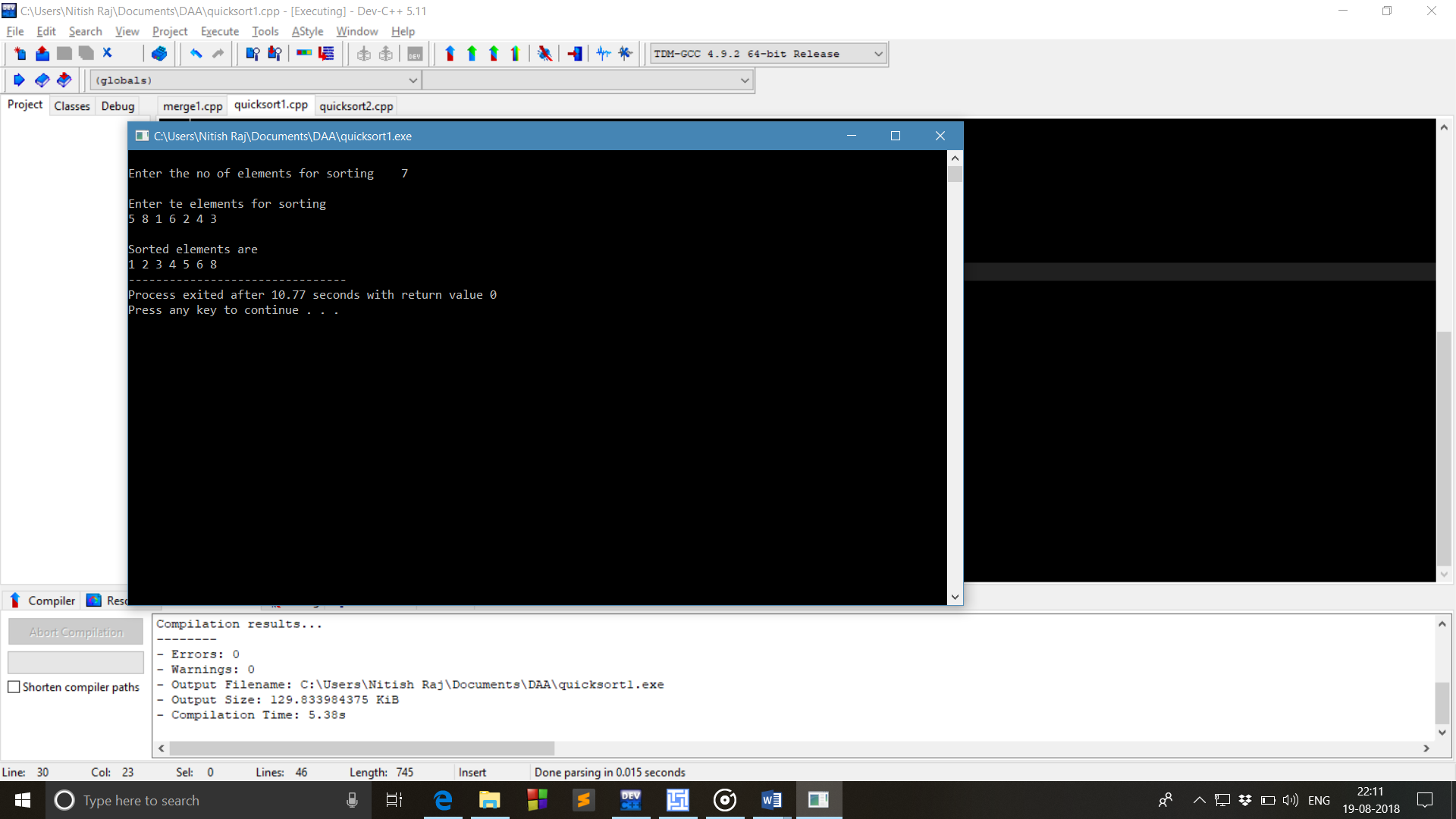
scanf("%d",&ar[i]);

quicksort(ar,0,n-1);

print(ar,n);

return 0;

}



1. **Implementation of Merge sort**

#include<stdio.h>

void merge(int a[],int l,int m,int h)

{

int i,j,k;

int n1=m-l+1;

int n2=h-m;

int ar1[n1],ar2[n2];

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

{

ar1[i]=a[l+i];

}

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

ar2[j]=a[m+1+j];

ar1[i]=9999;

ar2[j]=9999;

i=0;j=0;

for(k=l;k<=h;k++)

{

if(ar1[i]<=ar2[j])

a[k]=ar1[i++];

else

a[k]=ar2[j++];

}

}

void mergesort(int a[],int start,int end)

{

int mid;

if(start<end)

{

mid=(start+end)/2;

mergesort(a,start,mid);

mergesort(a,mid+1,end);

merge(a,start,mid,end);

}

}

int main()

{

int n,i;

printf("Enter no of elements for sorting: ");

scanf("%d",&n);

int ar[n];

printf("Enter the elements for sorting :\n");

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

scanf("%d",&a[i]);

mergesort(a,0,n-1);

printf("\nSorted elements are :");

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

printf("%d ",a[i]);

return 0;

}

