**4. Write an algorithm and program in to implement Selection sorting**

**i) Using arrays**

**ii) Using recursion**

**iii) Using recursion and linked list**

**i)Using array:-**

#include<stdio.h>

main()

{

int n,a[20],temp,i,j;

printf("\n enter no. of elements=");

scanf("%d",&n);

printf("\n enter values:");

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

{

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

}

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

for(j=i+1;j<n;j++)

{

if(a[i]>a[j]){

temp=a[i];

a[i]=a[j];

a[j]=temp;

}}}

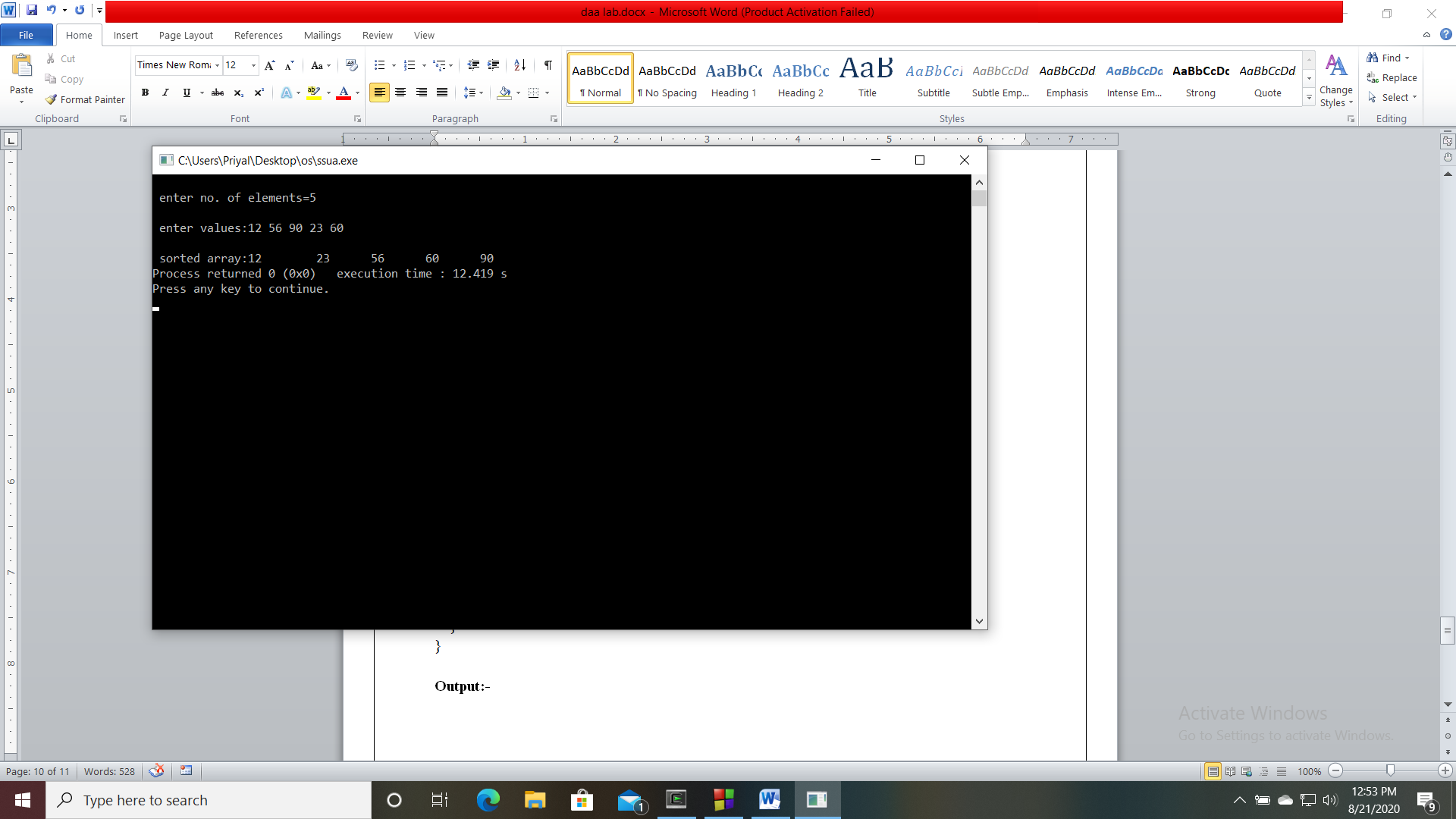
printf("\n sorted array:");

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

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

}}

**Output:-**



**(ii) Using recursion:-**

#include<stdio.h>

main()

{

int n,i,a[30];

printf("\n enter the no. of elements=");

scanf("%d",&n);

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

{

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

}

selectionsort(a,n,0);

printf(“\n sorted array=\n”);

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

{

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

}}

void selectionsort(int a[],int n,int x)

{

int j,temp;

int k=x;

for(j=x+1;j<n;j++)

{

if(a[k]>a[j])

{

temp=a[k];

a[k]=a[j];

a[j]=temp;

k=j;

}}

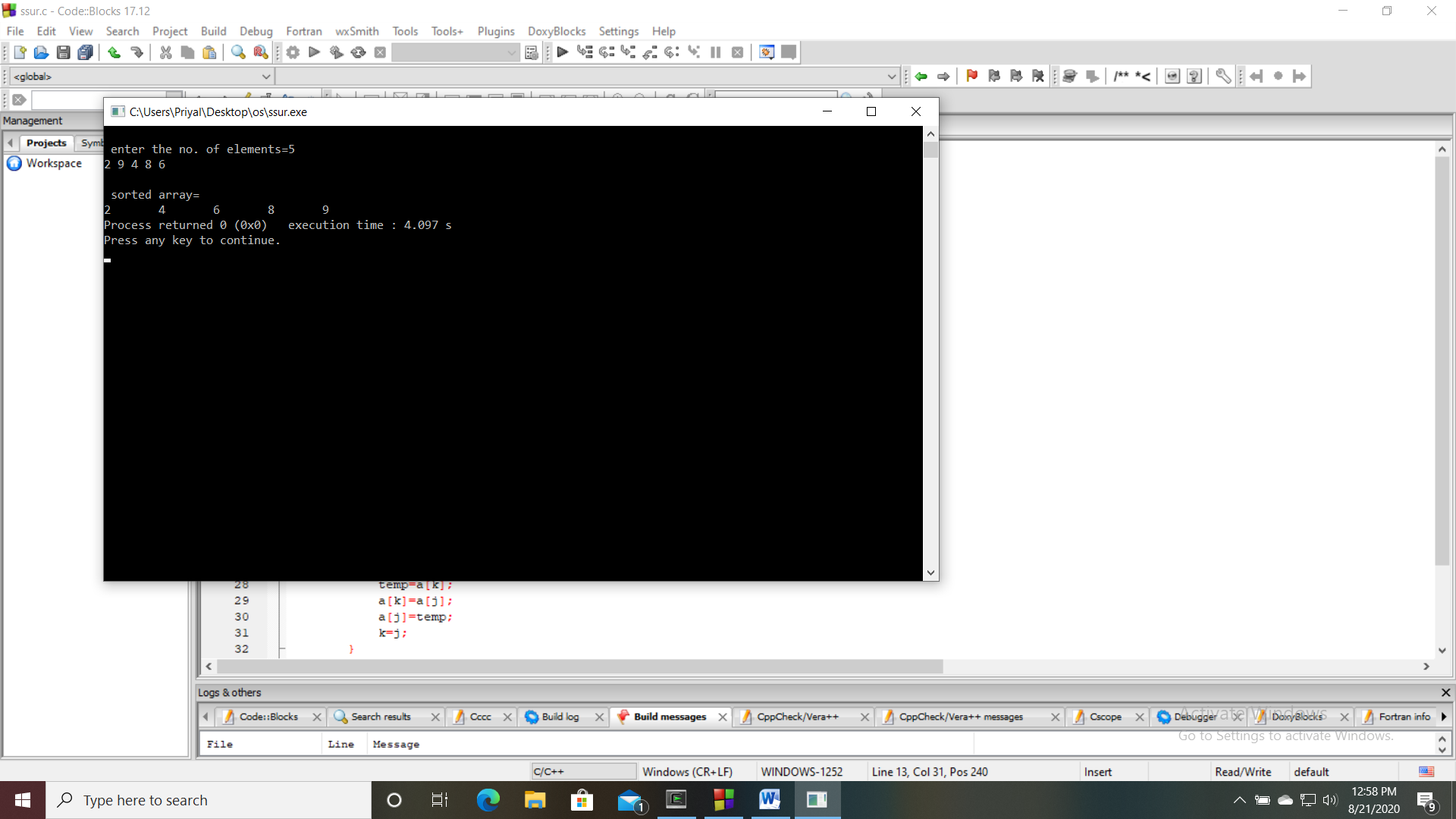
if(x+1<n)

{

selectionsort(a,n,x+1);

}}

**Output:-**



**(iii) Using recursion and linked list:-**

#include<stdio.h>

struct node

{

int data;

struct node \*link;

}\*start,\*node1,\*temp,\*ptr;

main()

{

create();

resort(start);

display();

}

void create()

{

int n;

printf("\n enter no. of nodes=");

scanf("%d",&n);

start=NULL;

while(n>=1)

{

node1=(struct node\*)malloc(sizeof(start));

node1->link=NULL;

scanf("%d",&node1->data);

if(start==NULL)

{

start=node1;

}

else{

temp=start;

while(temp->link!=NULL)

{

temp=temp->link;

}

temp->link=node1;

}

n--;

}}

void display()

{

if(start==NULL)

{

printf("\n link list is empty");

}

else

{

temp=start;

printf("\n sorted list=\n");

while(temp!=NULL)

{

printf("%d\t",temp->data);

temp=temp->link;

}}}

void resort(struct node \*head)

{

temp=head;

for(ptr=temp;ptr!=NULL;ptr=ptr->link)

{

if(ptr->data<temp->data)

{

int c=ptr->data;

ptr->data=temp->data;

temp->data=c;

}}

if(temp->link!=NULL)

{

resort(head->link);

}}

**Output:-**

