**PPS -LAB**

**Week-2**

**1.write a C program to find the area of a circle using the formula: Area= PI\*r2**

**C program:**

#include<stdio.h>

void main()

{

const float pi=3.1415;

float area,r;

printf("enter radius of the circle:");

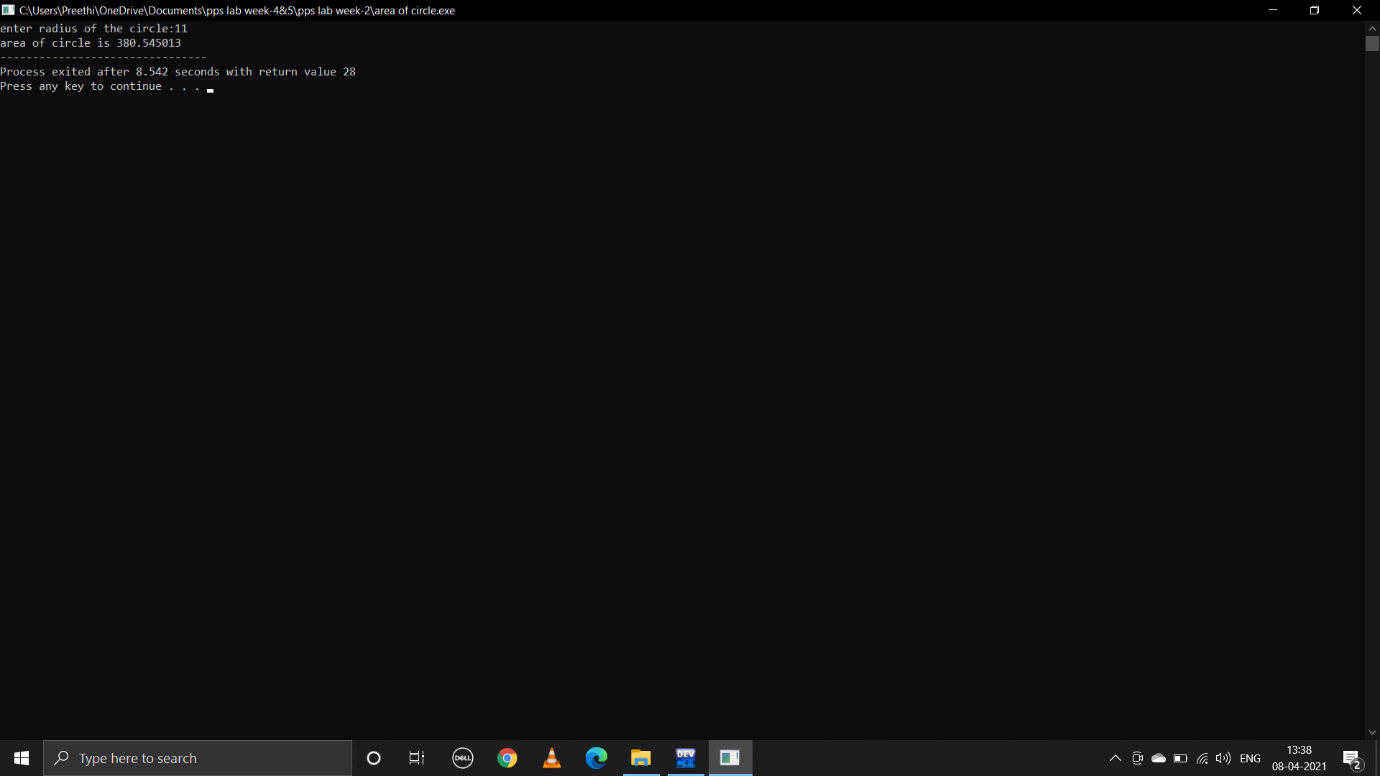
scanf("%f",&r);

area=pi\*r\*r;

printf("area of circle is %f",area);

}

**Output:**

****

**2.write a C program to find the area and volume of sphere.**

**Formulas are:Area=4\*PI\*R\*R Volume=4/3\*PI\*R\*R\*R**

**C program:**

#include<stdio.h>

void main()

{

const float pi=3.1415;

float r,area,volume;

printf("enter radius of the sphere:");

scanf("%f",&r);

area=4\*pi\*r\*r;

volume=(4\*pi\*r\*r\*r)/3;

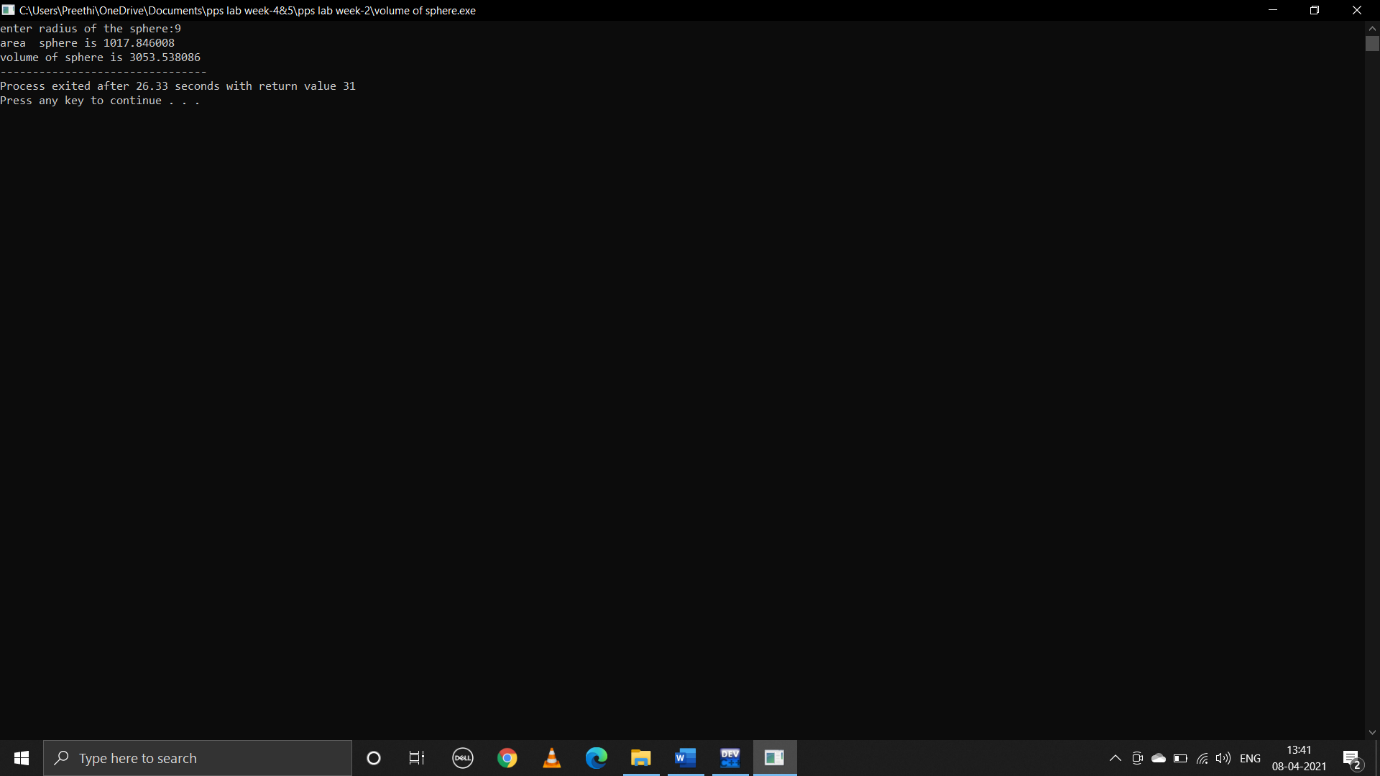
printf("area sphere is %f",area);

printf("\n");

printf("volume of sphere is %f",volume);

}

**Output:**

****

**3.write a C program to convert centigrade into farenheit.**

**Formula:c=(F-32)/1.8**

**C program:**

#include<stdio.h>

void main()

{

float c,f;

printf("enter temperature in celusis:");

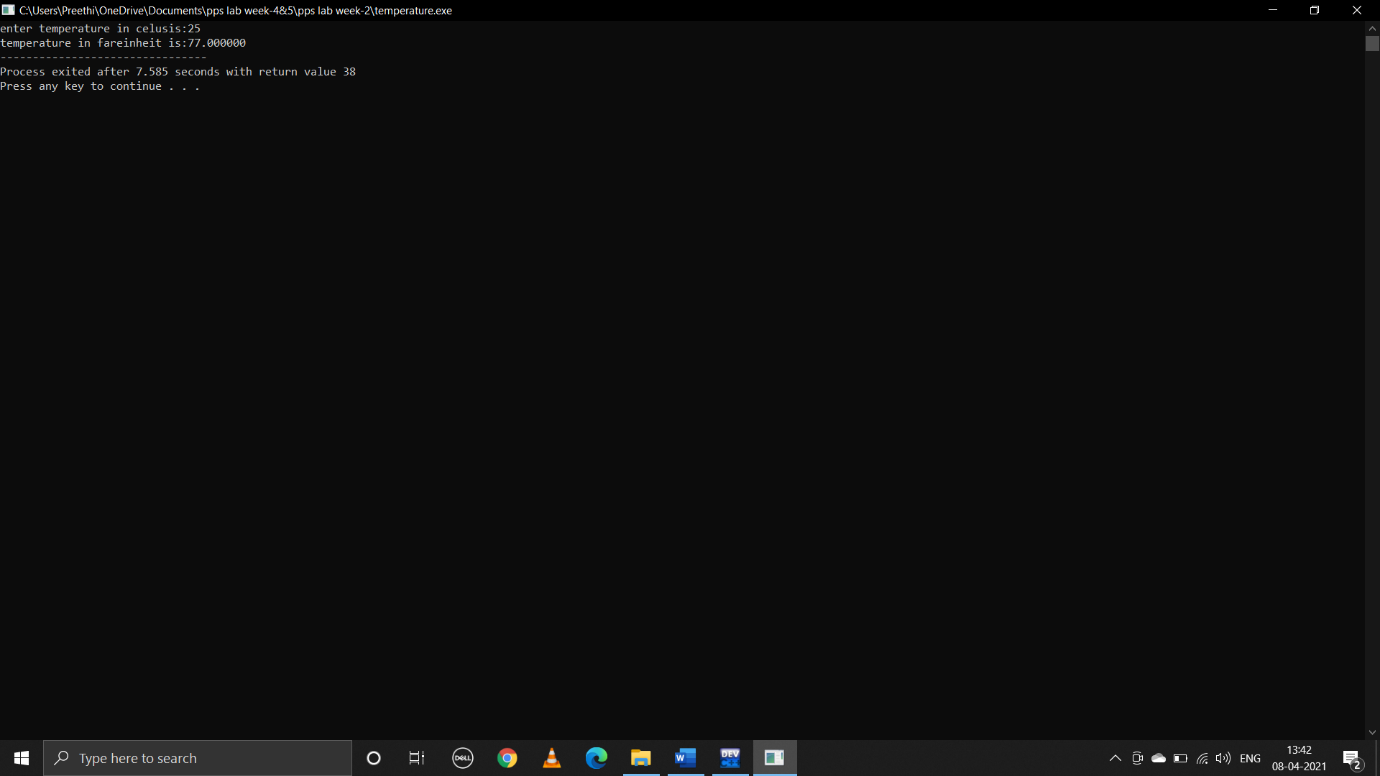
scanf("%f",&c);

f=(c\*1.8)+32;

printf("temperature in fareinheit is:%f",f);

}

**Output:**

****

**4.write a C program to read in two integers and display one as a percentage of the other.Typically your output should look like 20 is 50.00% of 40 assuming that the input numbers where 20 and 40 .Display the percentage correct to 2 decimal places.**

**C program:**

#include<stdio.h>

void main()

{

int a,b;

float c;

printf("enter a:");

scanf("%d",&a);

printf("enter b:");

scanf("%d",&b);

if(a<b)

{

c=(100\*a)/b;

printf("%d is %.2f%% of %d",a,c,b);

}

else if(a>b)

{

c=(100\*b)/a;

printf("%d is %.2f%% of %d",b,c,a);

}

else

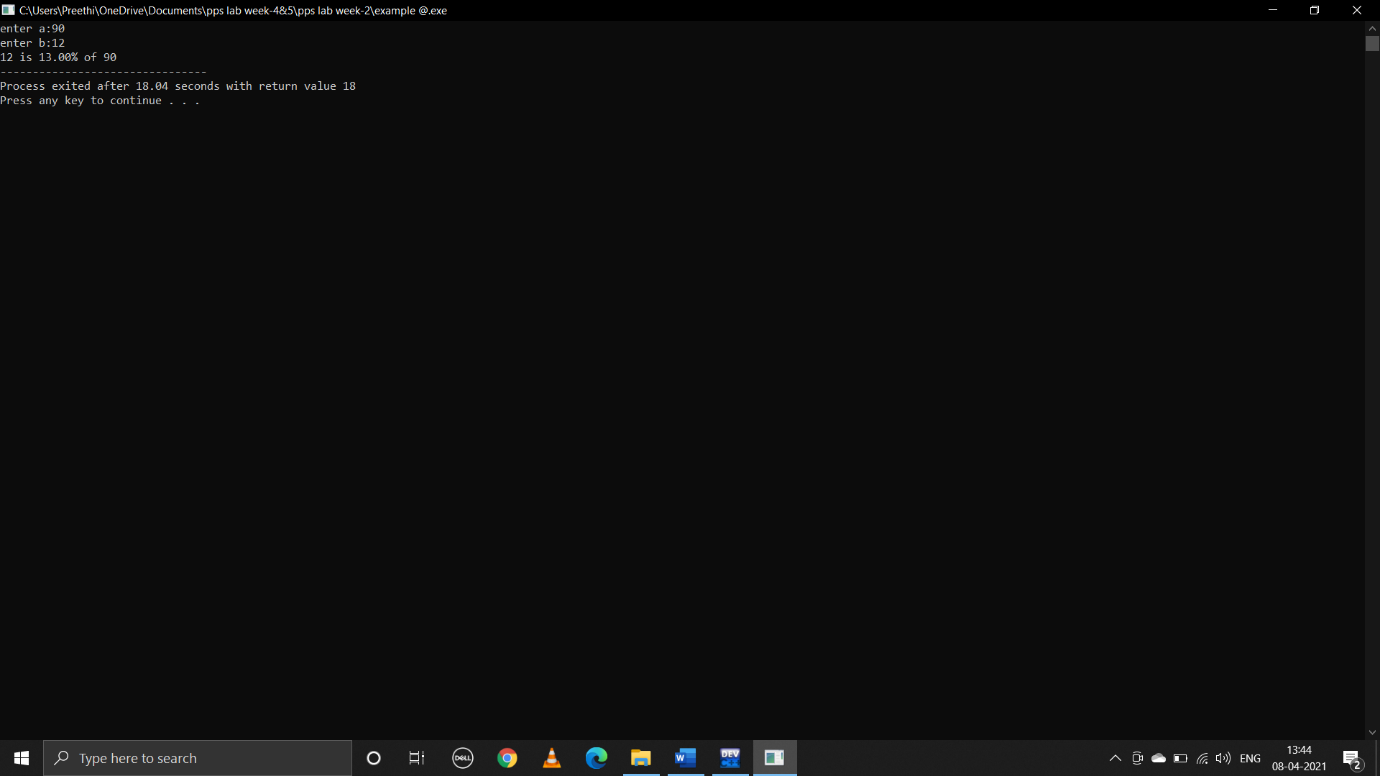
{

printf("both are equal");

}

}

**Output:**

****

**WEEK-3**

**1.write a C program to find the maximum from given three nos.**

**C program:**

#include<stdio.h>

void main()

{

int a,b,c;

printf("enter a,b,c values:");

scanf("%d%d%d",&a,&b,&c);

if((a>b)&&(a>c))

{

printf("%d is greater",a);

}

else if((b>a)&&(b>c))

{

printf("%d is greater",b);

}

else

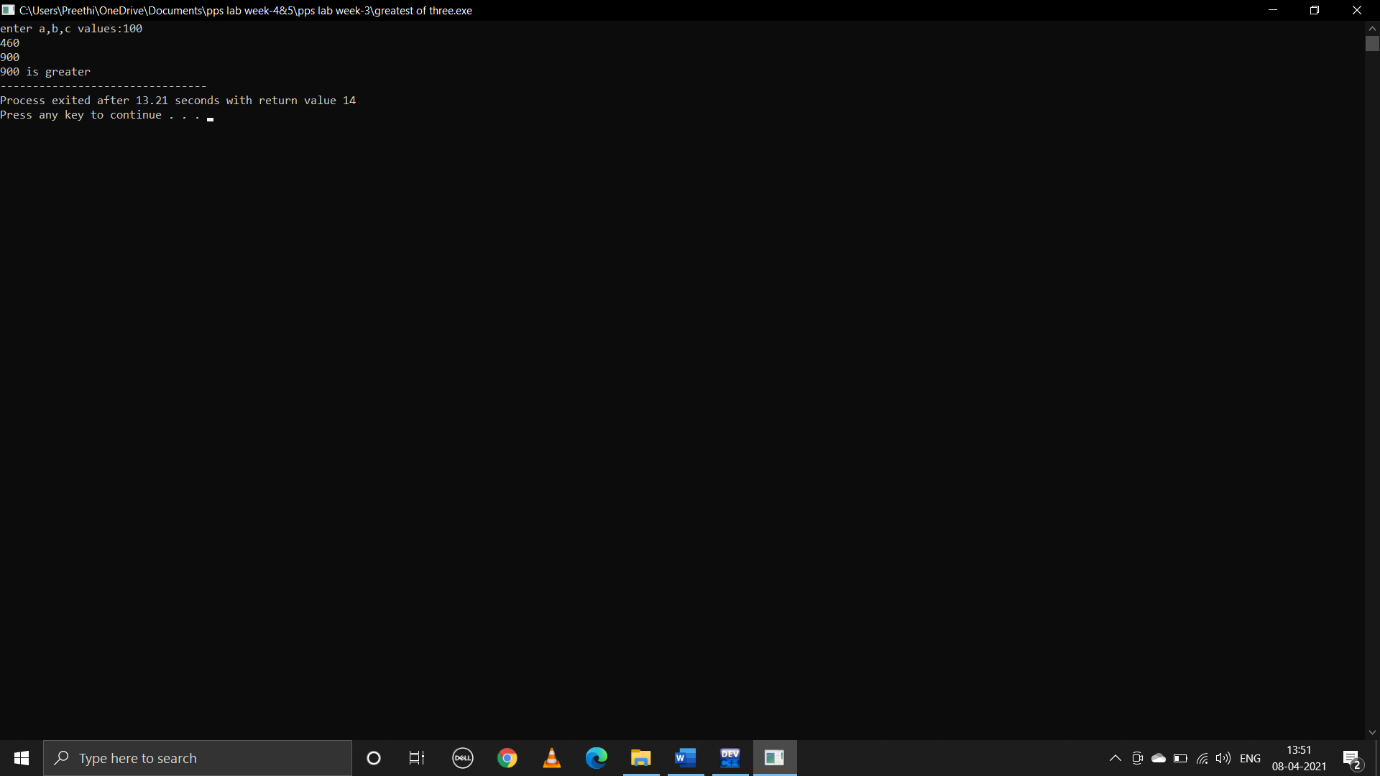
{

printf("%d is greater",c);

}

}

**Output:**

****

**2.write a C program to find that the accepted no is negative,positive or zero.**

**C program:**

#include<stdio.h>

void main()

{

int a;

printf("enter a value:");

scanf("%d",&a);

if(a>0)

{

printf("positive");

}

else if(a<0)

{

printf("negative");

}

else

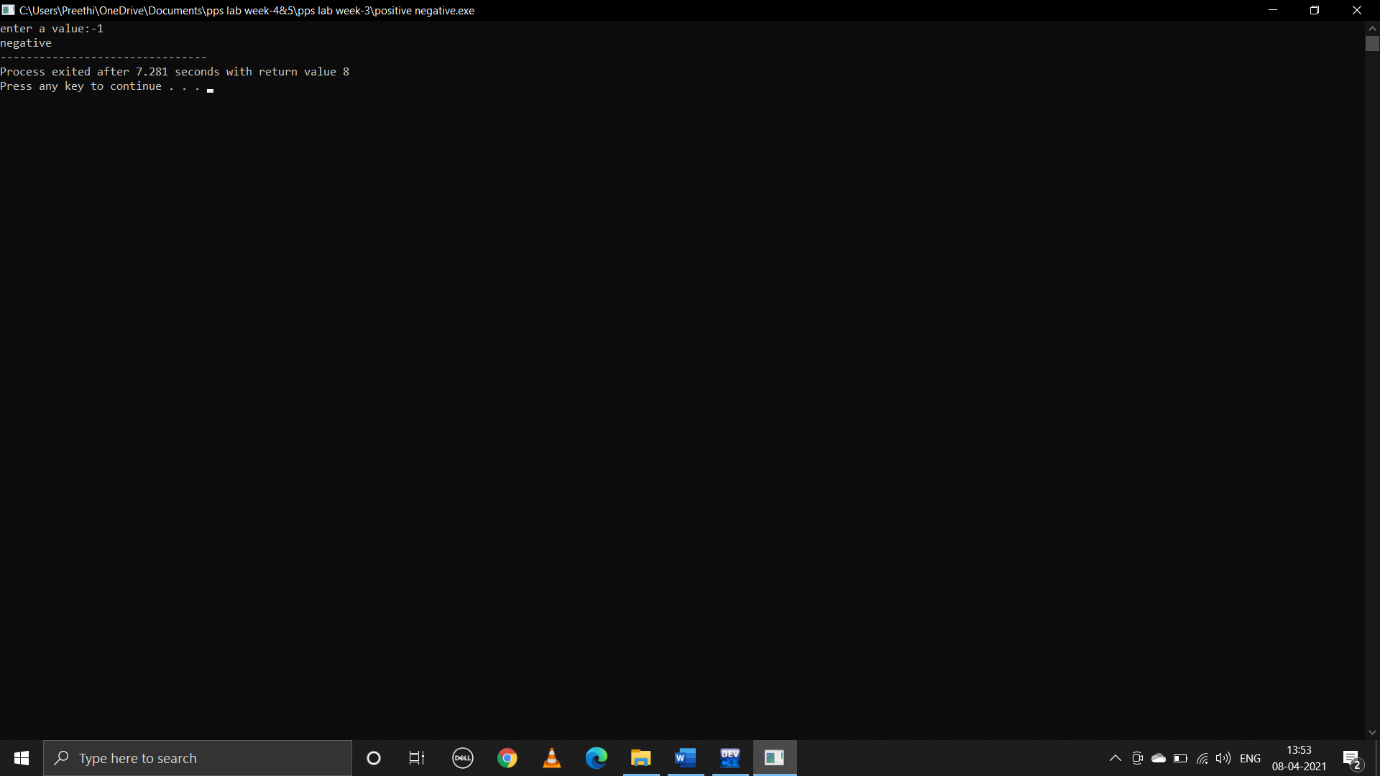
{

printf("0");

}

}

**Ouput:**

****

**3.write a program which reads two integer values.If the first is lesser print the message “up”.If the second is lesser,print the message “down”if they are equal,print the message “equal” if there is an error reading the data,print a message containing the word “Error”.**

**C program:**

#include<stdio.h>

void main()

{

int a,b;

printf("enter a,b values:");

scanf("%d %d",&a ,&b);

if(a<b)

{

printf("up");

}

else if(b<a)

{

printf("down");

}

else if(b==a)

{

printf("equal");

}

else

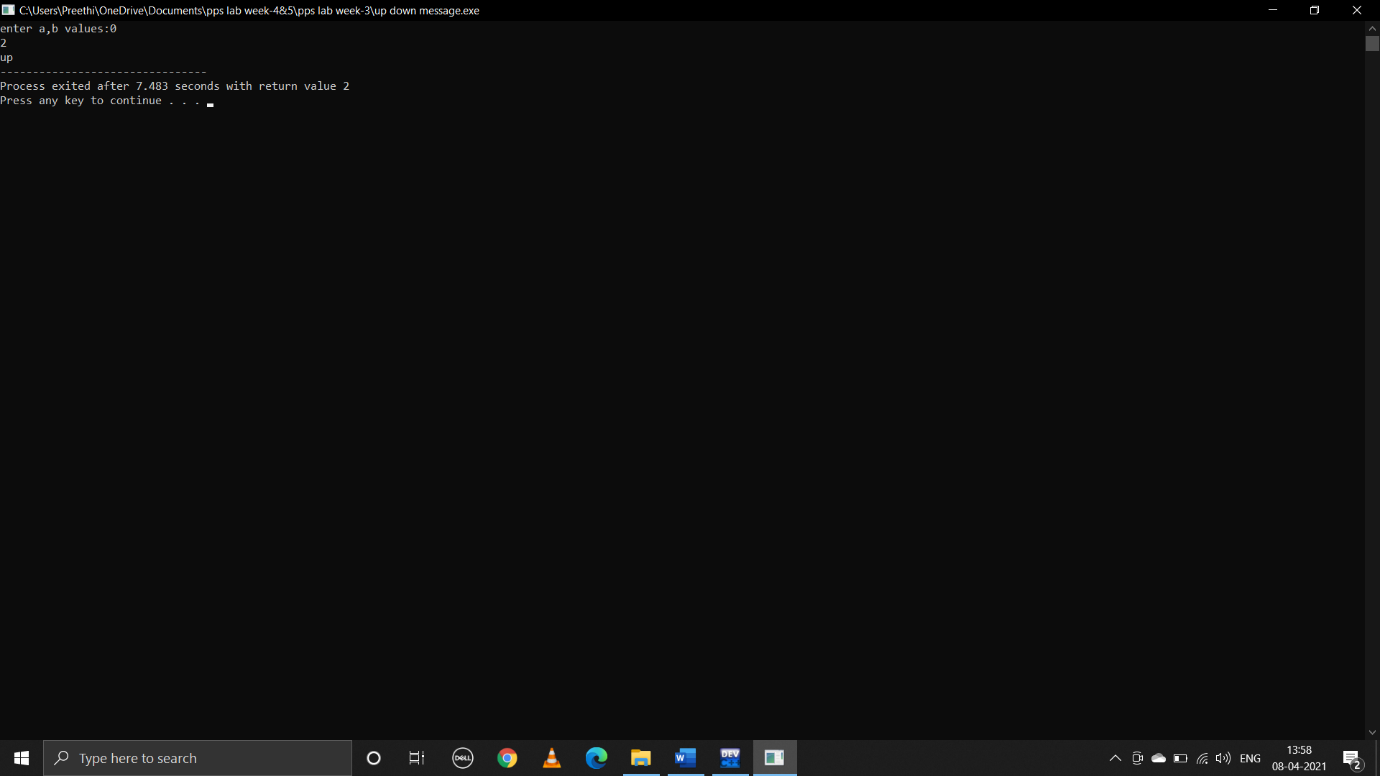
{

printf("error");

}

}

**Output:**

****

**4.write a C program that prints the given three integers in ascending order using if-else.**

**C program:**

#include<stdio.h>

void main()

{

int a,b,c;

printf("enter a,b,c values:");

scanf("%d%d%d",&a,&b,&c);

if((a<b)&&(a<c))

{

if(b<c)

{

printf("ascending order is:%d %d %d",a,b,c);

}

else

{

printf("ascending order is :%d %d %d",a,c,b);

}

}

else if((b<a)&&(b<c))

{

if(a<c)

{

printf("ascending order is:%d %d %d",b,a,c);

}

else

{

printf("ascending order is:%d %d %d",b,c,a);

}

}

else if((c<a)&&(c<b))

{

{

printf("ascending order is:%d %d %d",c,a,b);

}

else

{

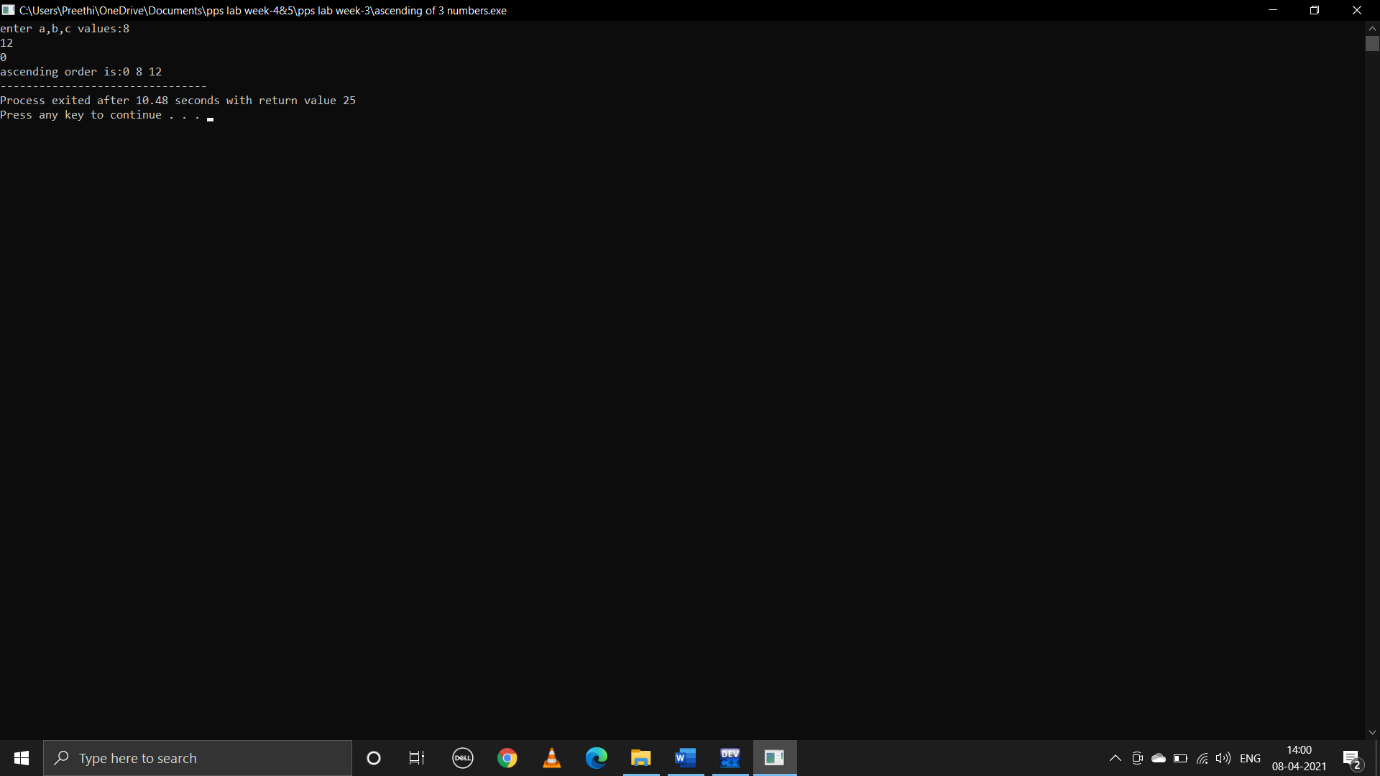
printf("ascending order is:%d %d %d",c,b,a);

}

}

}

**Output:**

****

**5.Given as input three integers representing a date as day,month,year,print the number day,month and year for the next day’s date.Typical input:”28 2 1992” Typical output: ” Date following 28:02:1992 is 29:02:1992”**

**C program:**

#include<stdio.h>

void main()

{

int d,m,y;

printf("enter date :");

scanf("%d",&d);

printf("enter month:");

scanf("%d",&m);

printf("enter year:");

scanf("%d",&y);

if(m==1||m==3||m==5||m==7||m==8||m==10||m==12)

{

if(d>=1&&d<31)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,d+1,m,y);

}

else if(d==31&&m!=12)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,1,m+1,y);

}

else if(d==31&&m==12)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,1,1,y+1);

}

}

else if(m==2)

{

if((y%4==0&&y%100!=0)||y%400==0)

{

if(d>=1&&d<29)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,d+1,m,y);

}

else if(d==29)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,1,m+1,y);

}

}

else

{

if(d>=1&&d<28)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,d+1,m,y);

}

else if(d==28)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,1,m+1,y);

}

}

}

else if(m==4||m==6||m==9||m==11)

{

if(d>=1&&d<30)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,d+1,m,y);

}

else if(d==30)

{

printf("date following %d:%d:%d is %d:%d:%d\n",d,m,y,1,m+1,y);

}

}

else

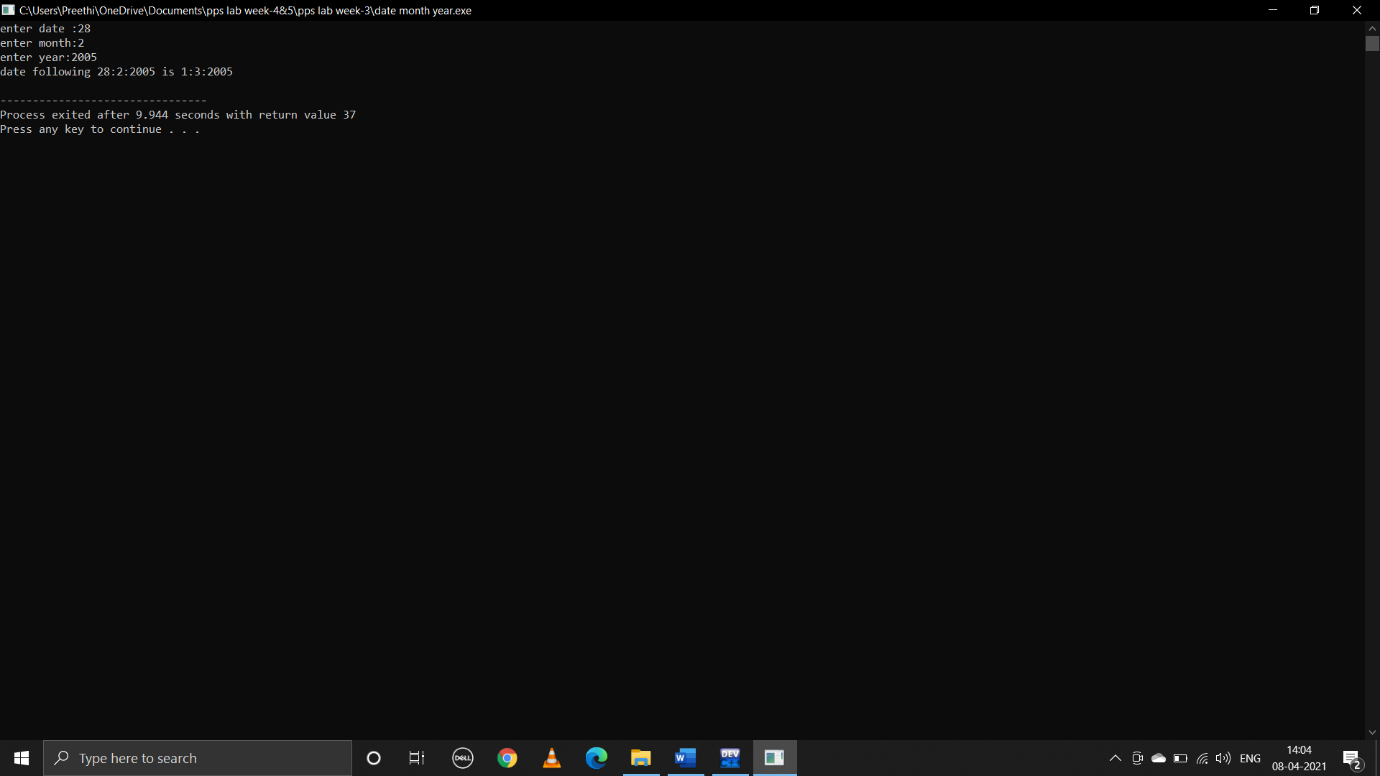
{

printf("entered wrong month");

}

}

**Output:**

****

**WEEK-4&5**

**1.write a C program to find the sum of first 100 odd nos. and even nos.**

**C program:**

#include<stdio.h>

void main()

{

Int i,odd=0,even=0;

for(i=1;i<=200;i=i+2)

{

even=even+i;

}

for(i=1;i<=200;i=i+2)

{

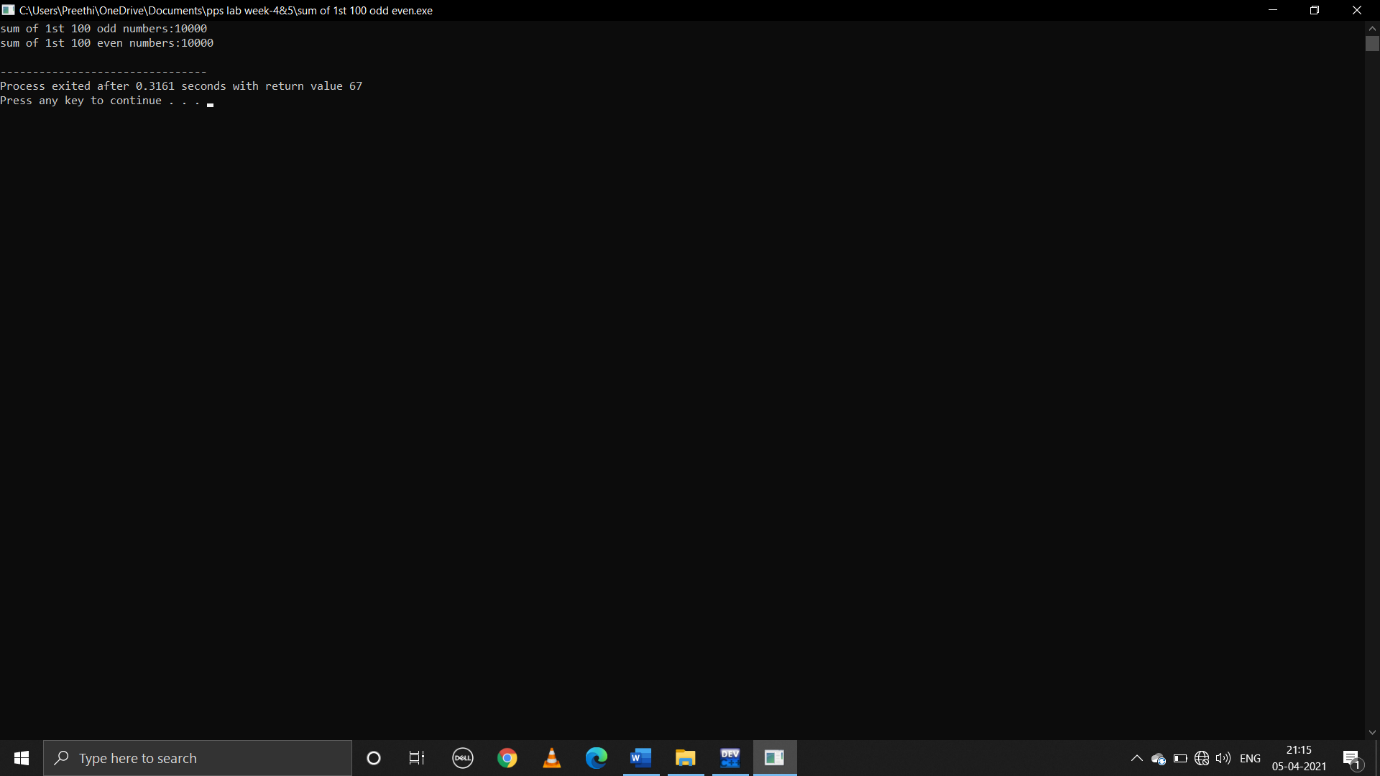
odd=odd+i;

}

printf("sum of 1st 100 odd numbers:%d\nsum of 1st 100 even numbers:%d\n",odd,even);

}

**Output:**

****

**2.write a C program to display first 100 prime nos.**

**C program:**

#include<stdio.h>

void main()

{

int n,i=3,count,c;

printf("enter the number of prime numbers required:");

scanf("%d",&n);

if(n>=1)

{

printf("first %d prime numbers are:",n);

printf("2 ");

}

for(count=2;count<=n;i++)

{

for(c=2;c<i;c++)

{

if(i%c==0)

break;

}

if(c==i)

{

printf("%d ",i);

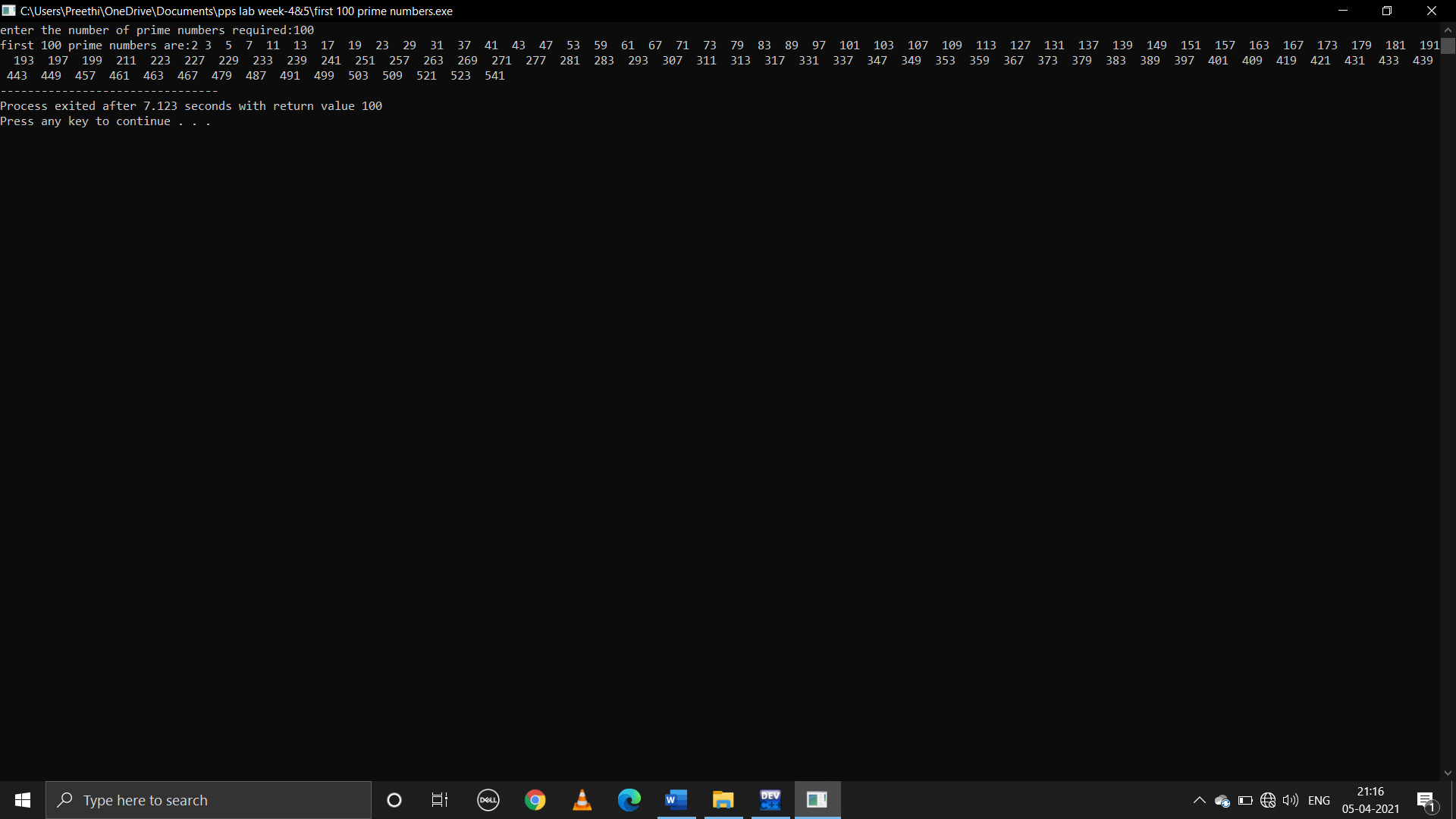
count++;

}

}

}

**Output:**

****

**3.write a C program to read in a three digit number produce following ouput (Assuming that the input is 347) 3 hundreds,4 tens,7 units**

**C program:**

#include<stdio.h>

void main()

{

int n,u,t;

printf("enter 3 digit number:\n");

scanf("%d",&n);

u=n%10;

n=n/10;

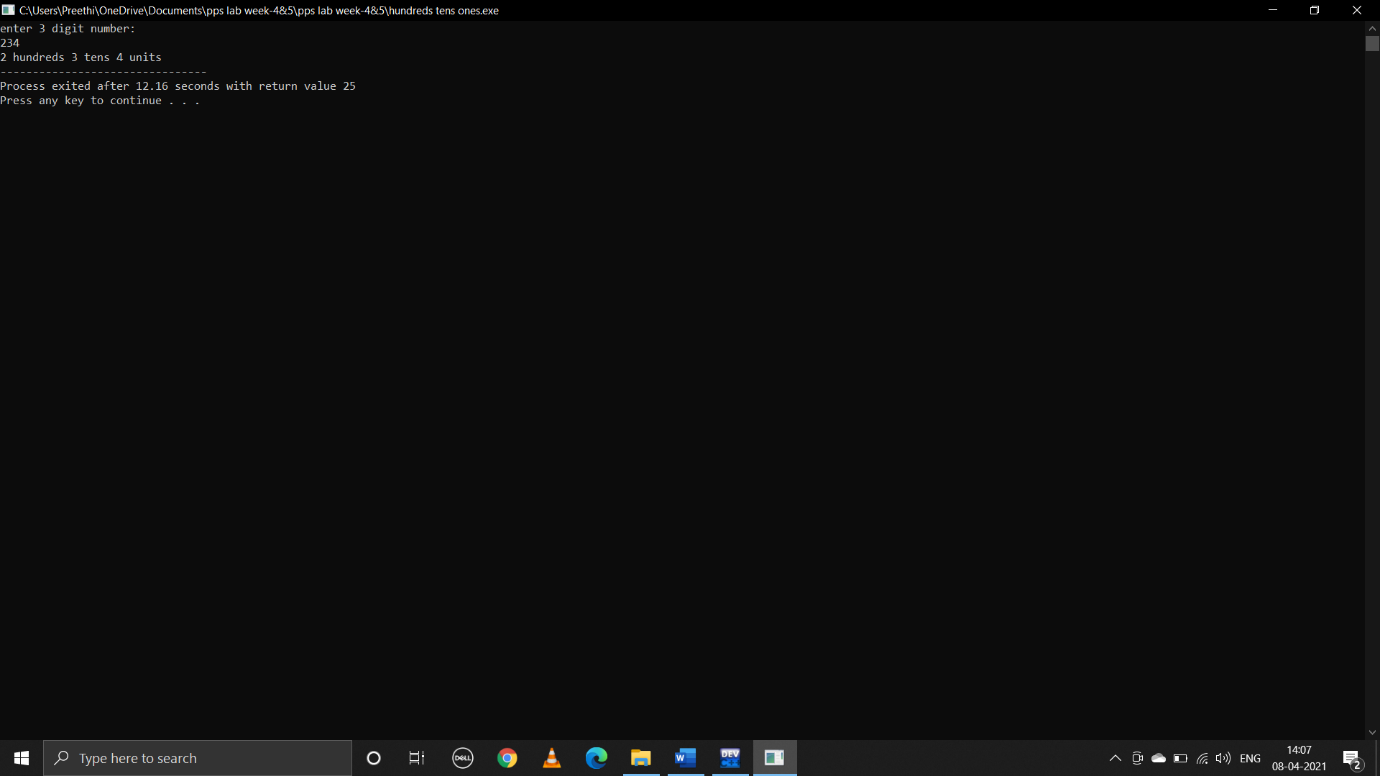
t=n%10;

n=n/10;

printf("%d hundreds %d tens %d units",n,t,u);

}

**Output:**

****

**4.write a C program to display Fibonacci series**

**C program:**

#include<stdio.h>

int main()

{

int n,i;

int nth;

int n1=0;

int n2=1;

printf("enter n:");

scanf("%d",&n);

printf("%d\n%d\n",n1,n2);

for(i=0;i<n-2;i++)

{

nth=n1+n2;

printf("%d\n",nth);

n1=n2;

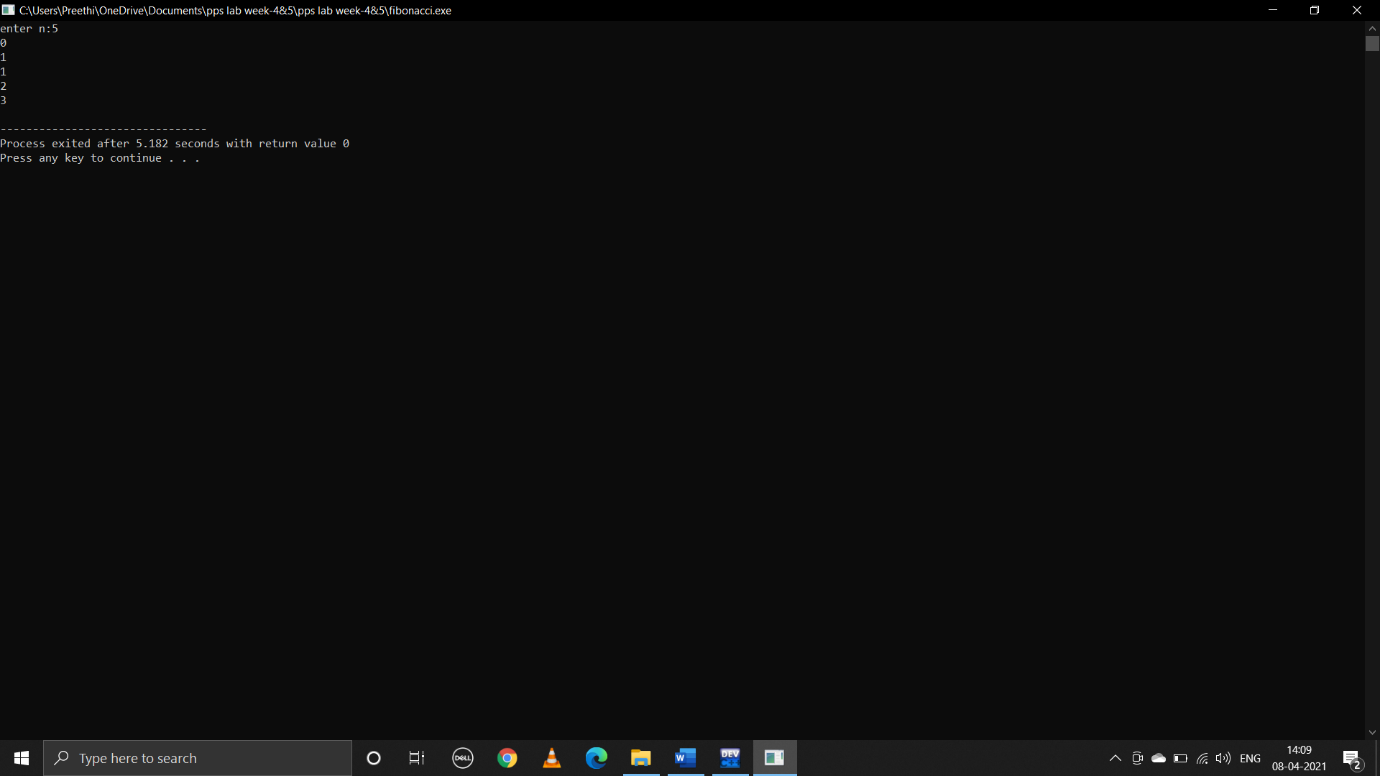
n2=nth;

}

return 0;

}

**Output:**

****

**5.write a C program to calculate the following**

**i.sum=1-x2/2!+x4/4!-x6/6!+x8/8!-x10/10!+……..,**

**C program:**

#include<stdio.h>

#include<math.h>

void main()

{

int n,x,p,fact,i,sign=-1;

float sum=1.0;

printf("enter n value:");

scanf("%d",&n);

printf("enter x value:");

scanf("%d",&x);

for(p=2;p<=n;p=p+2)

{

fact=1;

for(i=1;i<=p;i++)

{

fact=fact\*i;

}

sum=sum+(sign\*(pow(x,p))/fact);

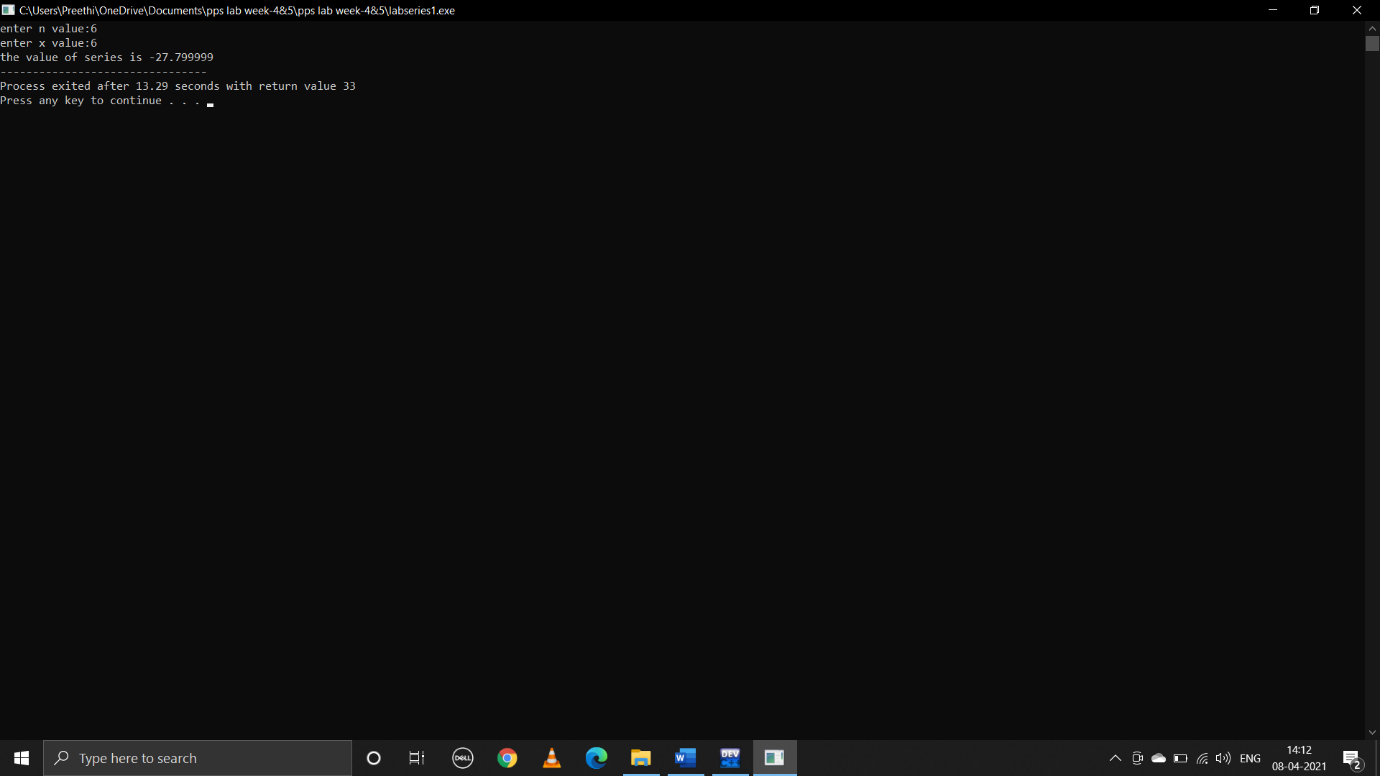
sign=sign\*-1;

}

printf("the value of series is %f",sum);

}

**Output:**

****

**ii.sum=x-x3/3!+x5/5!..............,**

**C program:**

#include<stdio.h>

#include<math.h>

void main()

{

int n,x,p,fact,i,sign=-1;

float sum=0.0;

printf("enter n value:");

scanf("%d",&n);

printf("enter x value:");

scanf("%d",&x);

for(p=1;p<=n;p=p+2)

{

fact=1;

for(i=1;i<=p;i++)

{

fact=fact\*i;

}

sign=sign\*-1;

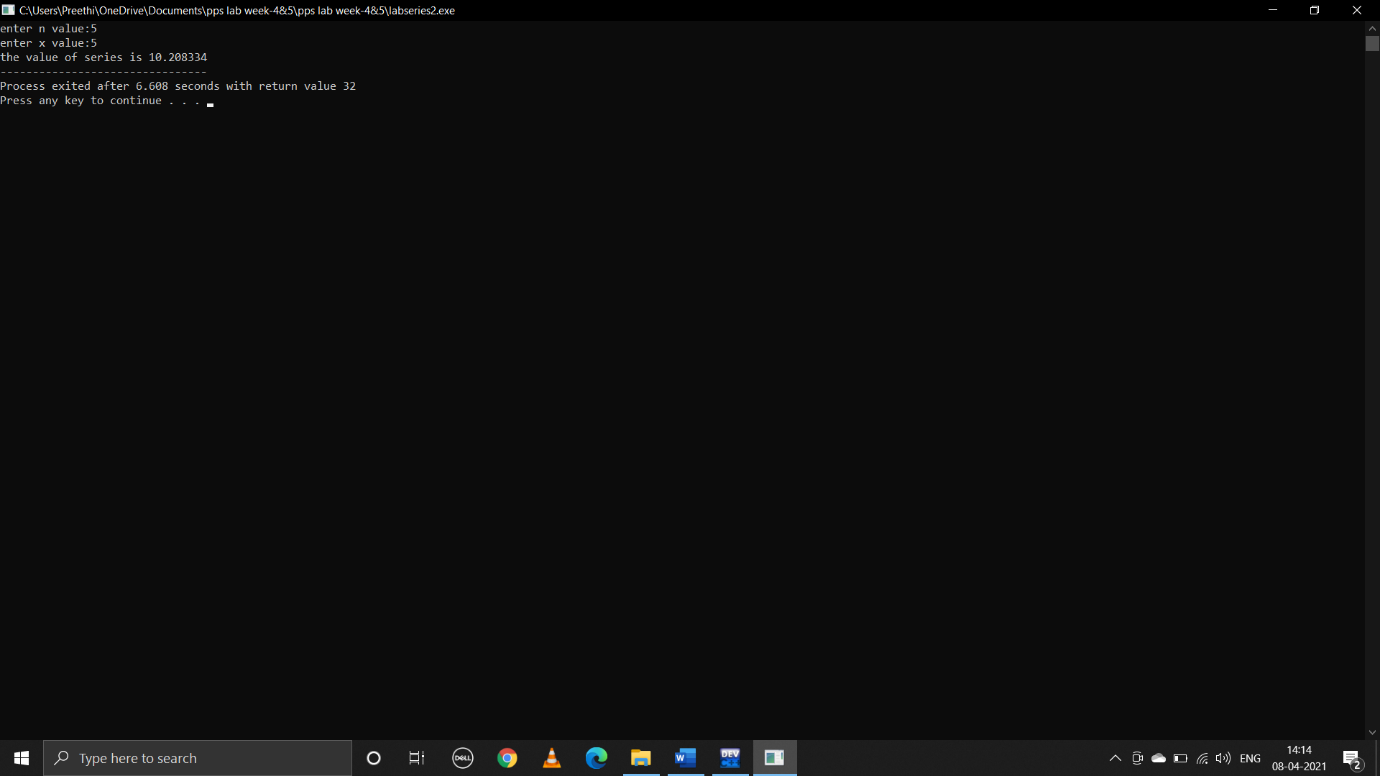
sum=sum+(sign\*(pow(x,p))/fact);

}

printf("the value of series is %f",sum);

}

**Output:**

****

**iii.sum=1+x/1!+x^2/2!+x^3/3!............,**

**C program:**

#include<stdio.h>

#include<math.h>

void main()

{

int n,x,p,fact,i,sign=1;

float sum=1.0;

printf("enter n value:");

scanf("%d",&n);

printf("enter x value:");

scanf("%d",&x);

for(p=1;p<n;p++)

fact=1;

for(i=1;i<=p;i++)

{

fact=fact\*i;

}

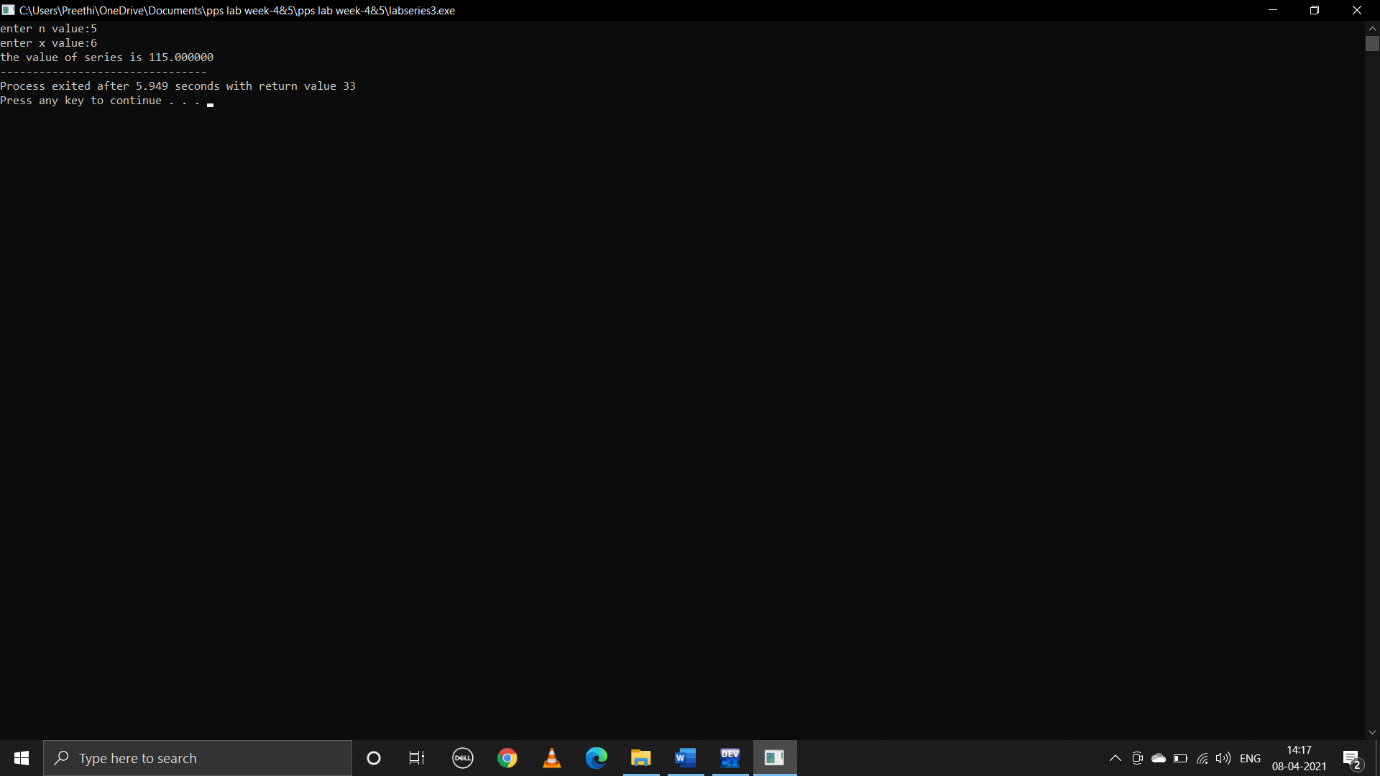
sum=sum+(sign\*(pow(x,p))/fact);

}

printf("the value of series is %f",sum);

}

**Output:**

****

**6.write a C program to find the roots of a quadratic equation.**

**C program:**

#include<stdio.h>

#include<math.h>

void main()

{

float a,b,c,d,imaginary;

float root1,root2;

printf("enter a,b,c values:");

scanf("%f%f%f",&a,&b,&c);

d=b\*b-4\*a\*c;

if(d>0)

{

root1=(-b+sqrt(d))/(2\*a);

root2=(-b-sqrt(d))/(2\*a);

printf(" two distinct real roots of quadratic eqn are:%.2f\n %.2f\n",root1,root2);

}

else if(d<0)

{

root1=root2=-b/(2\*a);

imaginary=sqrt(-d)/(2\*a);

printf(" two distinct imaginary roots of quadratic eqn are:%.2f+i%.2f\n %.2f-i%.2f\n",root1, imaginary,root2,imaginary);

}

else if(d==0)

{

root1=(-b)/(2\*a);

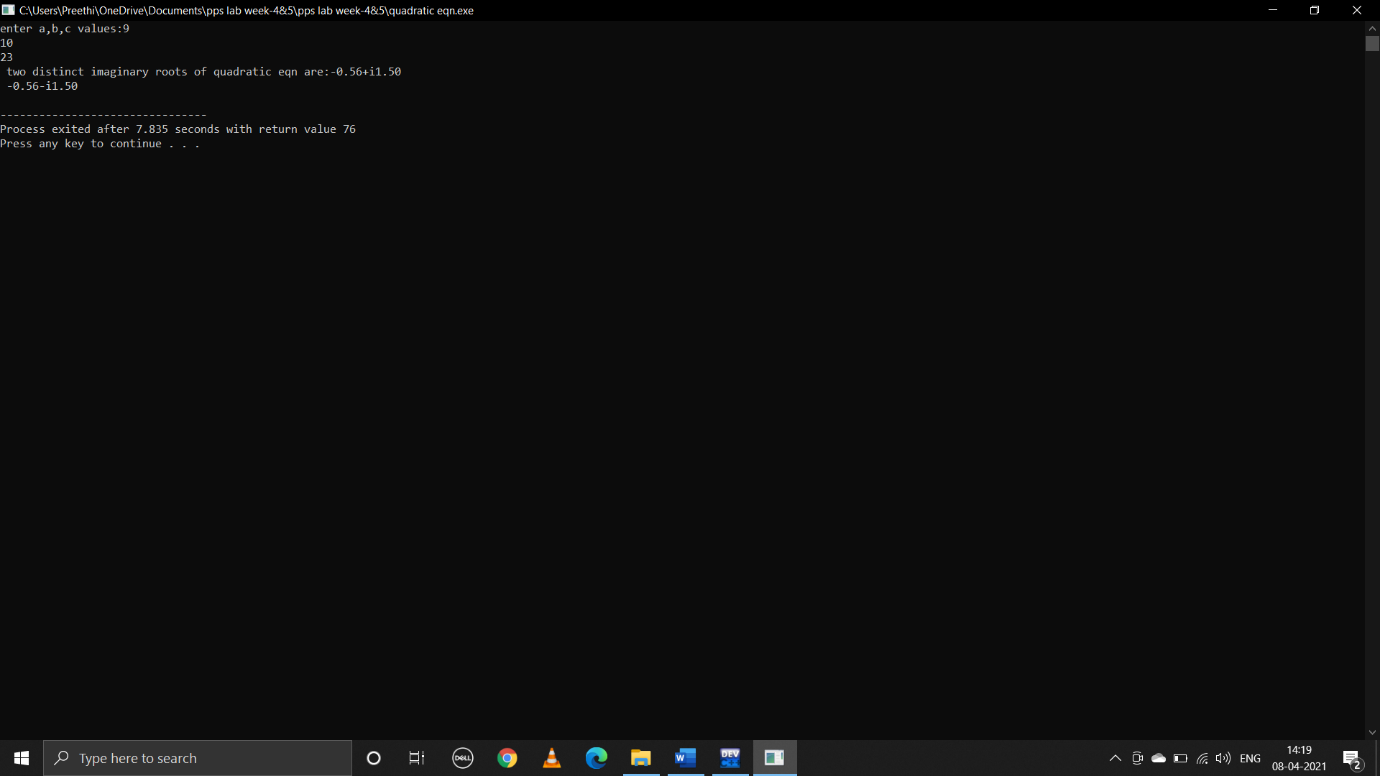
root1=root2;

printf("two equal real roots of the quadratic eqn are:%.2f\n %.2f\n",root1,root2);

}

}

**Output:**

****

**WEEK-6**

**1.C program that reads N integers numbers and arrange them in ascending order using Bubble sort**

**C program:**

#include<stdio.h>

void bubblesort(int arr[],int size);

main()

{

int n,i;

printf("enter size of an array:");

scanf("%d",&n);

int a[n];

printf("enter elements in array:\n");

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

{

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

}

printf("unsorted array is:\n");

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

{

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

}

printf("\n");

bubblesort(a,n);

}

void bubblesort(int arr[],int size)

{

int temp,i,j;

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

{

for(j=0;j<size-1;j++)

{

if(arr[j]>arr[j+1])

{

temp=arr[j];

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

arr[j+1]=temp;

}

}

printf("array status after pass %d\n",i);

printarray(arr,size);

printf("\n");

}

}

void printarray(int arr[],int size)

{

int i;

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

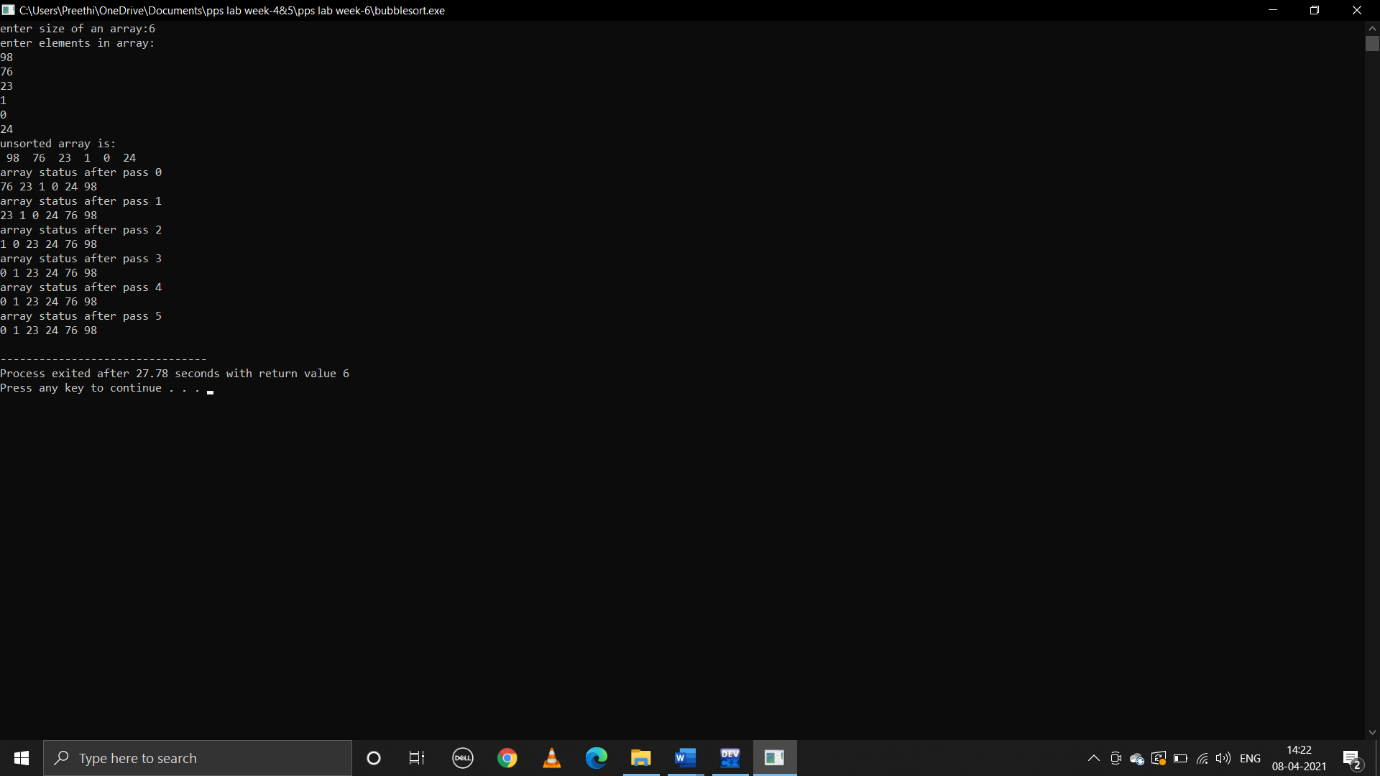
{

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

}

}

**Output:**

****

**2.C program that reads N integer numbers and arrange them in ascending order using merge sort**

**C program:**

#include<stdio.h>

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

{

int n1=q-p+1;

int n2=r-q;

int l[n1],m[n2];

int i,j,k;

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

{

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

}

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

{

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

}

i=0;

j=0;

k=p;

while(i<n1 && j<n2)

{

if(l[i]<=m[j])

{

arr[k]=l[i];

i++;

}

else

{

arr[k]=m[j];

j++;

}

k++;

}

while(i<n1)

{

arr[k]=l[i];

i++;

k++;

}

while(j<n2)

{

arr[k]=m[j];

j++;

k++;

}

printarray(arr,n);

}

void mergesort(int arr[],int l,int r,int n)

{

if(l<r)

{

int m=(l+r)/2;

mergesort(arr,l,m,n);

mergesort(arr,m+1,r,n);

merge(arr,l,m,r,n);

}

}

void printarray(int arr[],int size)

{

static int i=0;

printf("array status after pass %d:",i);

i++;

int j;

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

{

printf("%d ",arr[j]);

}

printf("\n");

}

int main()

{

int n;

printf("enter size of an array:");

scanf("%d",&n);

int i,arr[n];

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

{

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

}

printf("unsorted array is:\n");

printarray(arr,n);

mergesort(arr,0,n-1,n);

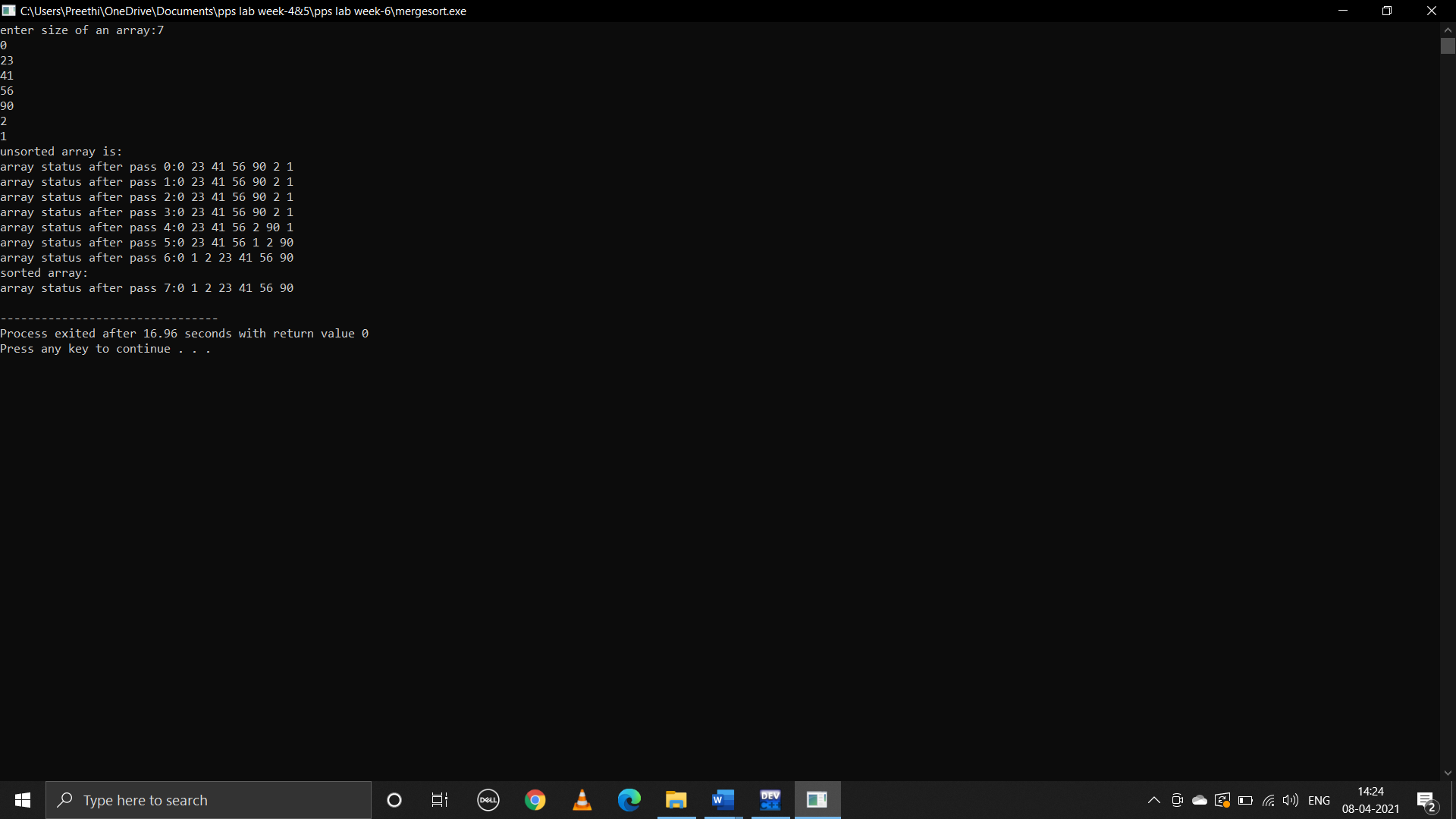
printf("sorted array:\n");

printarray(arr,n);

return 0;

}

**Output:**

****

**3.C program that reads N integer numbers and arrange them in ascending order using Quick Sort**

**C program:**

#include<stdio.h>

void quicksort(int a[],int first,int last,int n)

{

int i,j,pivot,temp;

if(first<last)

{

pivot=first;

i=first;

j=last;

while(i<j)

{

while(a[i]<=a[pivot]&&i<last)

i++;

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

j--;

if(i<j)

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

temp=a[pivot];

a[pivot]=a[j];

a[j]=temp;

printarray(a,n);

printf("\n");

quicksort(a,0,j-1,n);

quicksort(a,j+1,last,n);

}

}

int main()

{

int n,i;

printf("enter size of array:");

scanf("%d",&n);

int a[n];

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

{

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

}

printf("before sorted:\n");

printarray(a,n);

quicksort(a,0,n-1,n);

printf("after sorted:\n");

printarray(a,n);

return 0;

}

void printarray(int a[],int last)

{

static int i=0;

printf("array status after pass %d:",i);

i++;

int j;

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

{

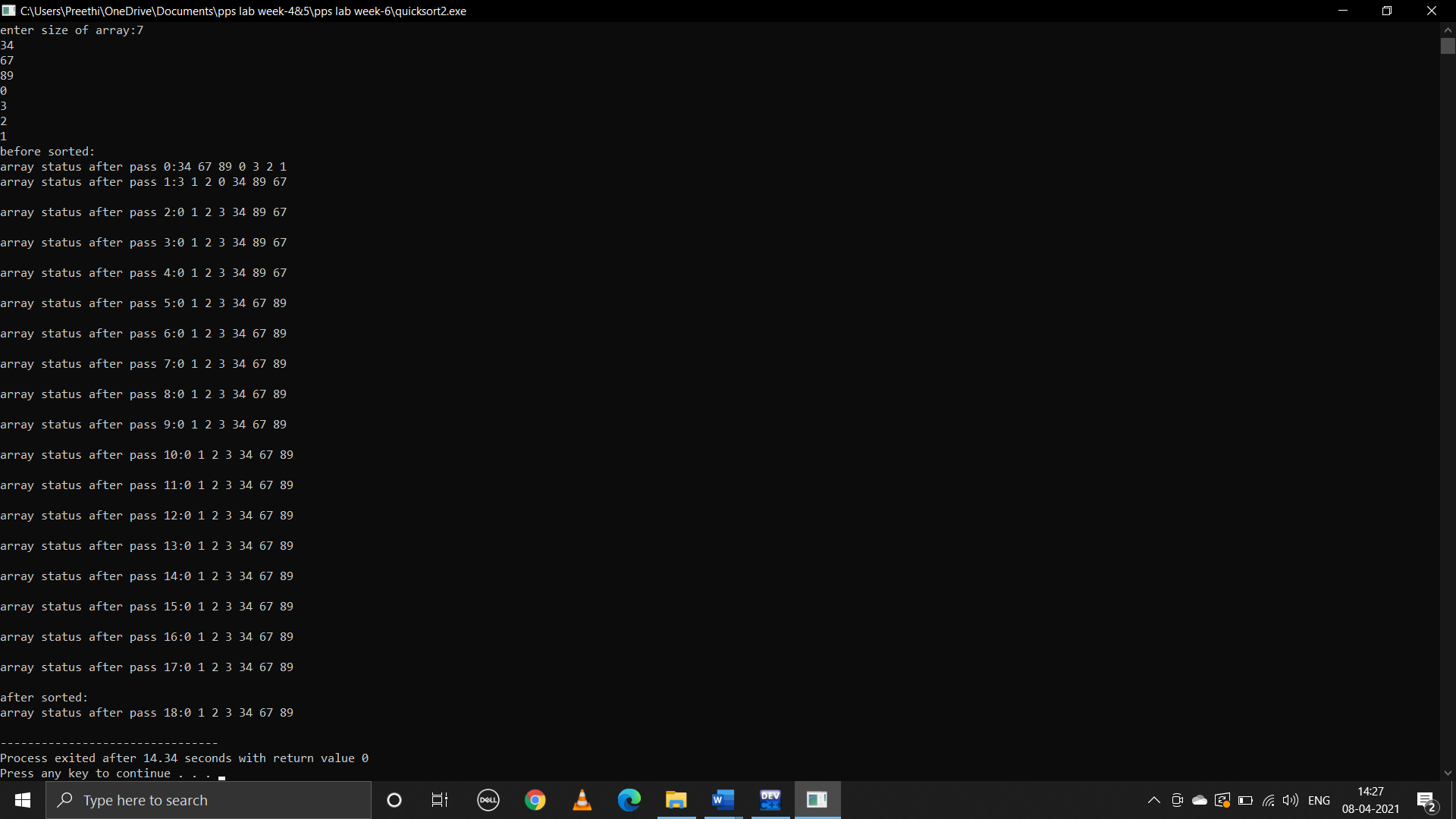
printf("%d ",a[j]);

}

printf("\n");

}

**Output:**

****

**WEEK-7**

**1.Write a C program to perform the basic Matrix operations**

**i)addition**

**C program:**

#include<stdio.h>

void main()

{

int rows,columns,i,j;

printf("enter no.of rows:\n");

scanf("%d",&rows);

printf("enter no.of columns:\n");

scanf("%d",&columns);

printf("enter elements in 1st matrix");

int a[rows][columns];

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

{

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

{

printf("element-%d%d:",i,j);

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

}

}

int b[rows][columns];

printf("enter elements in 2nd matrix");

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

{

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

{

printf("element-%d%d:",i,j);

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

}

}

int sum[rows][columns];

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

{

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

{

sum[i][j]=a[i][j]+b[i][j];

}

}

printf("sum of two matrices is:");

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

{

printf("\n");

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

{

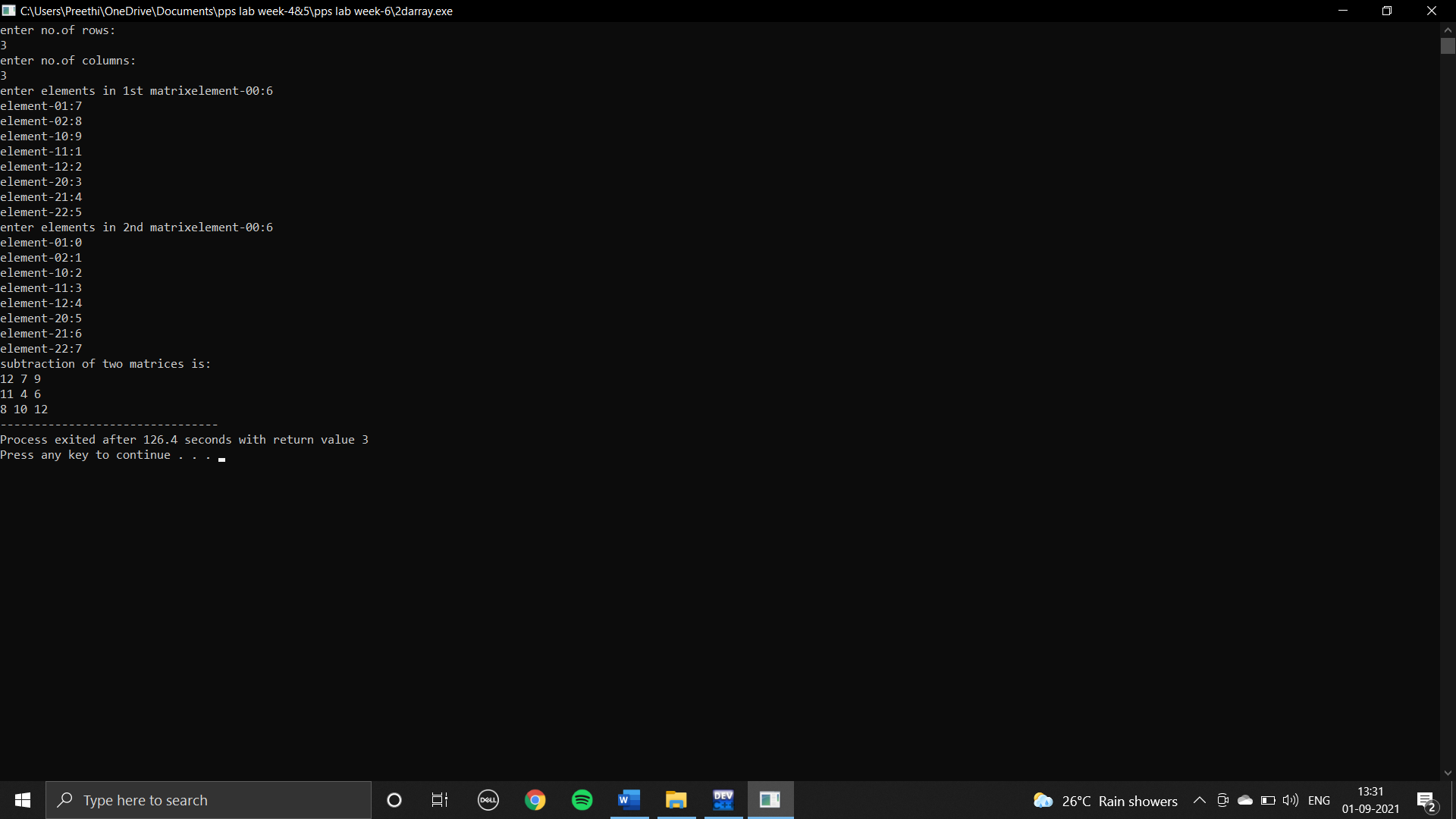
printf("%d ",sum[i][j]);

}

}

}

**Output:**

****

**ii)subtraction:**

**c program:**

#include<stdio.h>

void main()

{

int rows,columns,i,j;

printf("enter no.of rows:\n");

scanf("%d",&rows);

printf("enter no.of columns:\n");

scanf("%d",&columns);

printf("enter elements in 1st matrix");

int a[rows][columns];

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

{

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

{

printf("element-%d%d:",i,j);

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

}

}

int b[rows][columns];

printf("enter elements in 2nd matrix");

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

{

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

{

printf("element-%d%d:",i,j);

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

}

}

int sub[rows][columns];

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

{

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

{

sub[i][j]=a[i][j]-b[i][j];

}

}

printf("subtraction of two matrices is:");

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

{

printf("\n");

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

{

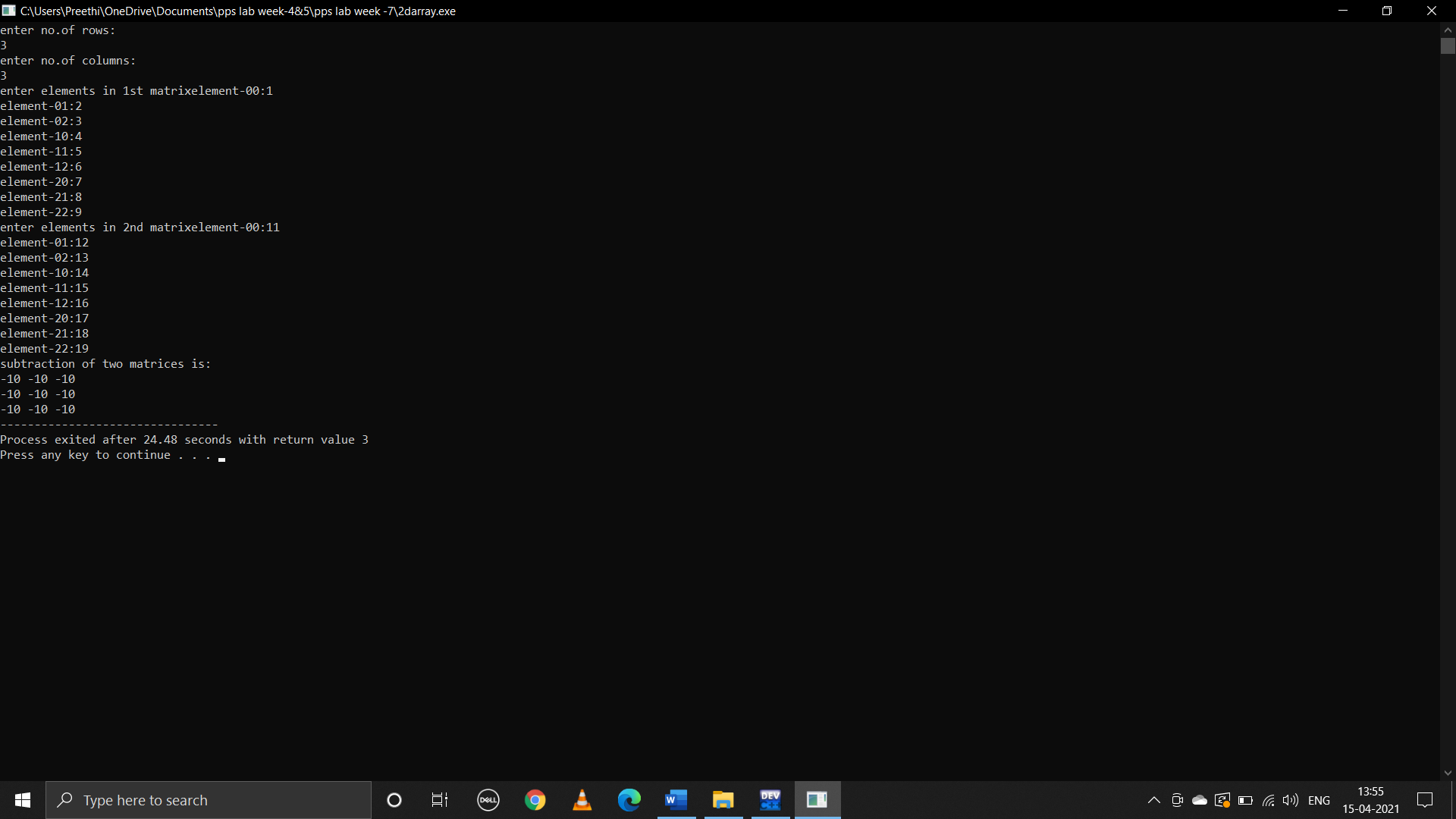
printf("%d ",sub[i][j]);

}

}

}

**Output:**

****

**iii)Multiplication**

**C program:**

#include<stdio.h>

void main()

{

int i,j,r1,c1,r2,c2,sum=0,k,multiply[100][100];

printf("enter rows for 1st matrix:");

scanf("%d",&r1);

printf("enter columns for 1st matrix:");

scanf("%d",&c1);

int a[r1][c1];

for(i=0;i<r1;i++)

{

for(j=0;j<c1;j++)

{

printf("enter %d%d element:",i,j);

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

}

}

for(i=0;i<r1;i++)

{

printf("\n");

for(j=0;j<c1;j++)

{

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

}

}

printf("\n");

printf("enter rows for 2nd matrix:");

scanf("%d",&r2);

printf("enter columns for 2nd matrix:");

scanf("%d",&c2);

int b[r2][c2];

for(i=0;i<r2;i++)

{

for(j=0;j<c2;j++)

{

printf("enter %d%d element:",i,j);

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

}

}

for(i=0;i<r2;i++)

{

printf("\n");

for(j=0;j<c2;j++)

{

printf("%d ",b[i][j]);

}

}

printf("\n");

if(c1!=r2)

{

printf("multiplication is not possible");

}

else

{

for(i=0;i<r1;i++)

{

for(j=0;j<c2;j++)

{

for(k=0;k<r2;k++)

{

sum=sum+a[i][k]\*b[k][j];

}

multiply[i][j]=sum;

sum=0;

}

}

printf("\n");

printf("product of entered matrices:\n");

for(i=0;i<r1;i++)

{

printf("\n");

for(j=0;j<c2;j++)

{

printf("%d ",multiply[i][j]);

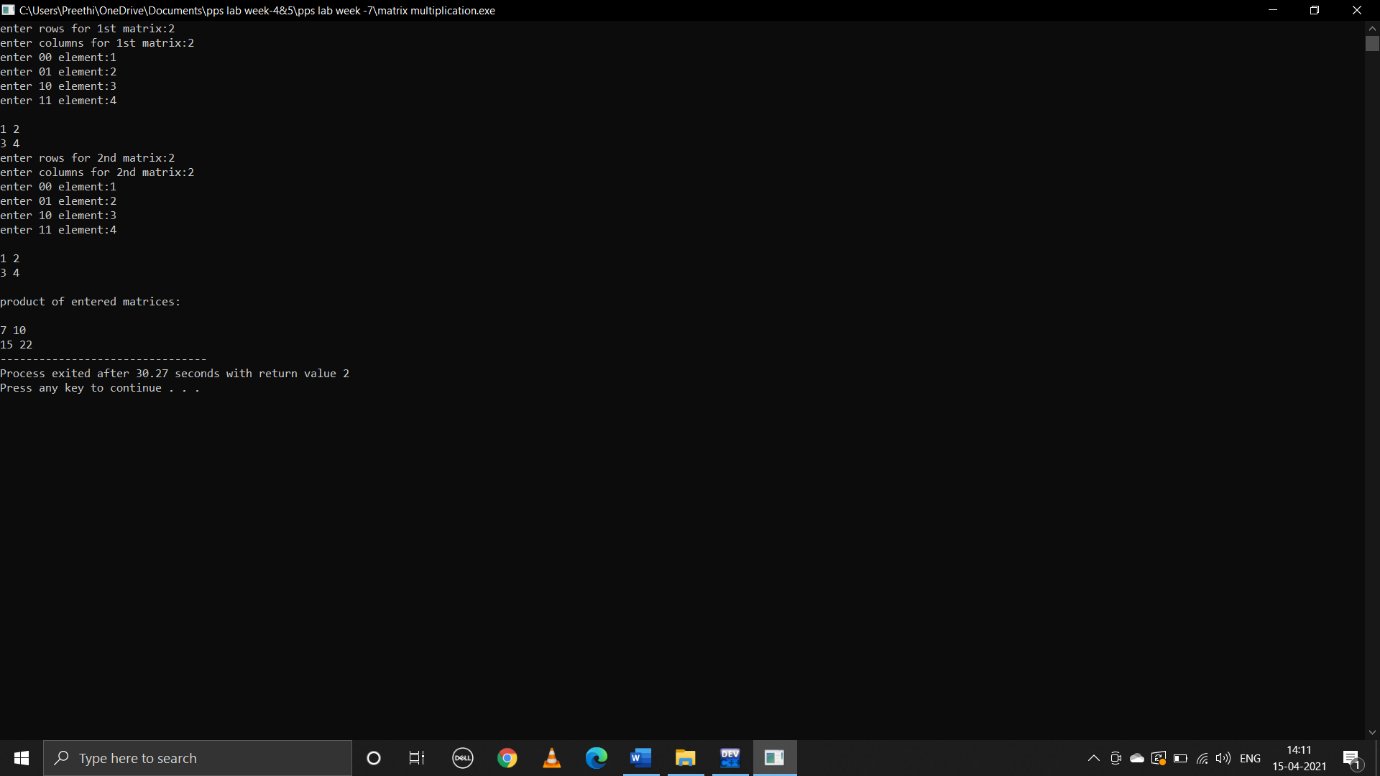
}

}

}

}

**Output:**

****

**iv)Transpose**

**C program:**

#include<stdio.h>

void main()

{

int r1,c1,i,j;

printf("enter no.of rows of 1st matrix:");

scanf("%d",&r1);

printf("enter no.of columns of 1st matrix:");

scanf("%d",&c1);

int a[r1][c1];

printf("enter elements of matrix a\n");

for(i=0;i<r1;i++)

{

for(j=0;j<c1;j++)

{

printf("enter element-%d%d:",i,j);

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

}

}

for(i=0;i<r1;i++)

{

printf("\n");

for(j=0;j<c1;j++)

{

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

}

}

printf("\n");

int trans[c1][r1];

for(i=0;i<r1;i++)

{

for(j=0;j<c1;j++)

trans[j][i]=a[i][j];

}

printf("transpose of matrix is:\n");

for(i=0;i<r1;i++)

{

printf("\n");

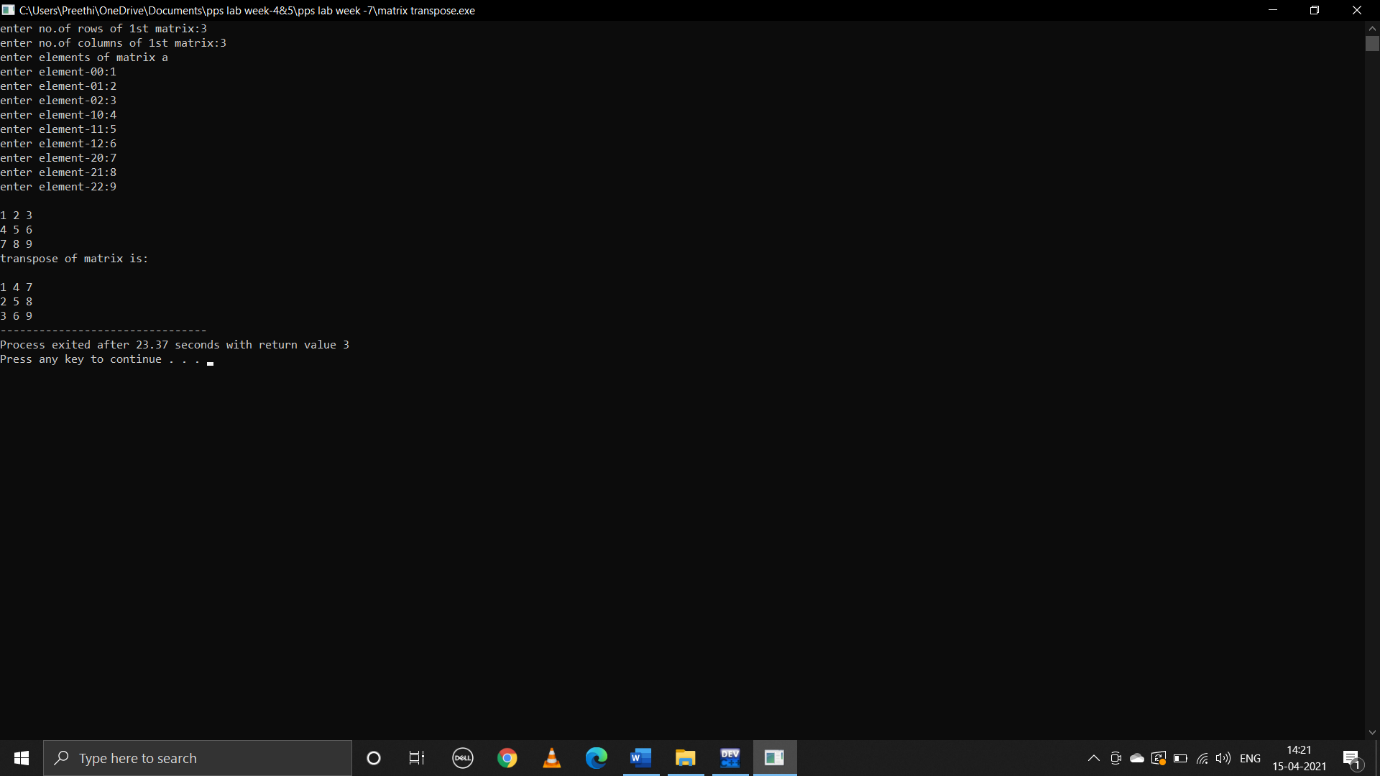
for(j=0;j<c1;j++)

printf("%d ",trans[i][j]);

}

}

**Output:**

****

**2.write a C program to determine if the given string is a palindrome or not**

**C program:**

#include<stdio.h>

#include<string.h>

void main()

{

char string[1000];

printf("enter any string:");

gets(string);

puts(string);

int i,len,count=0;

len=strlen(string);

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

{

if(string[i]==string[len-i-1])

count=count+1;

}

if(count==len)

{

printf("palindrome");

}

else

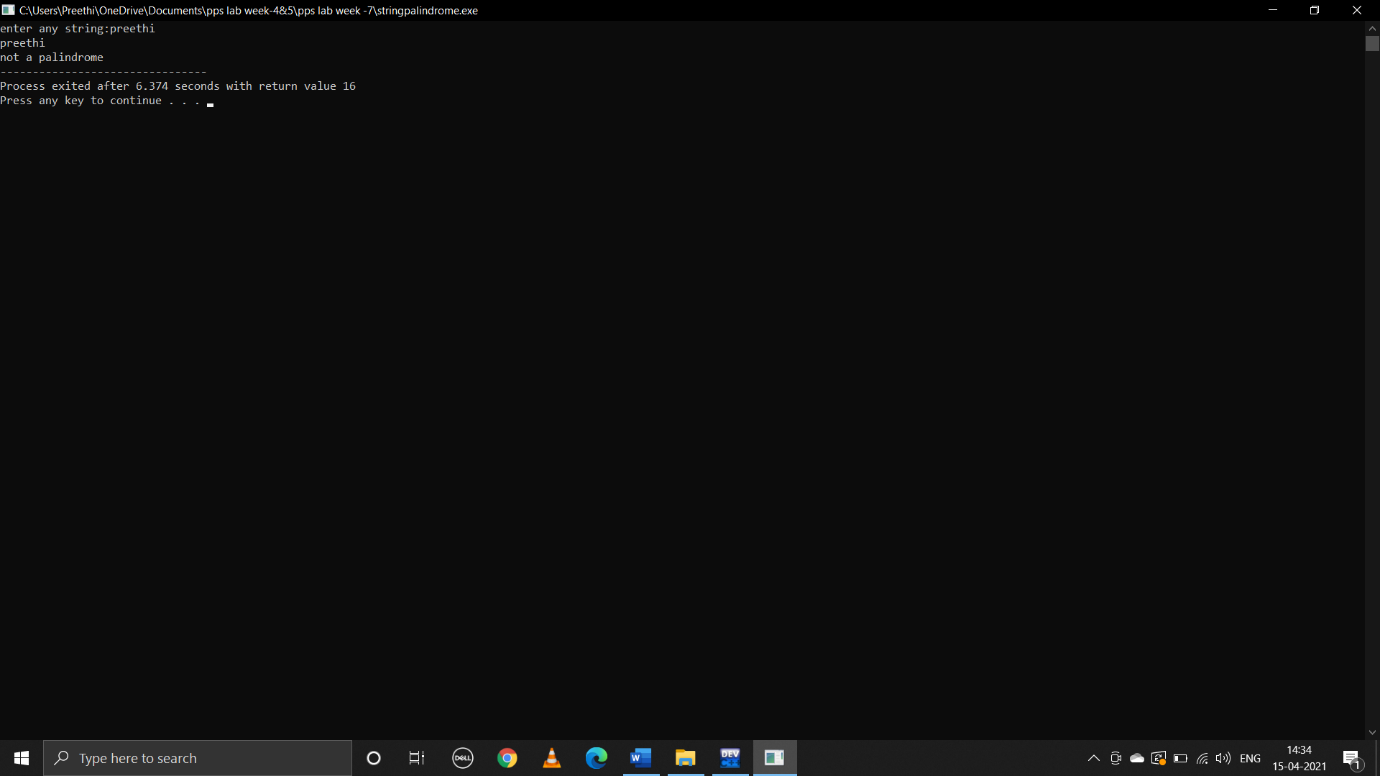
{

printf("not a palindrome");

}

}

**Output:**



**3.write a C program to count the lines,words and characters in a given text**

**C program:**

#include<stdio.h>

#include<string.h>

void main()

{

char s[100];

int i,line=0,word=0,ch=0;

printf("enter paragraph terminated with ~:");

scanf("%[^~]",s);

for(i=0;s[i]!='\0';i++)

{

if(s[i]=='\n')

{

line++;

word++;

}

else

{

if(s[i]==' '||s[i]=='\t')

{

word++;

ch++;

}

else

{

ch++;

}

}

}

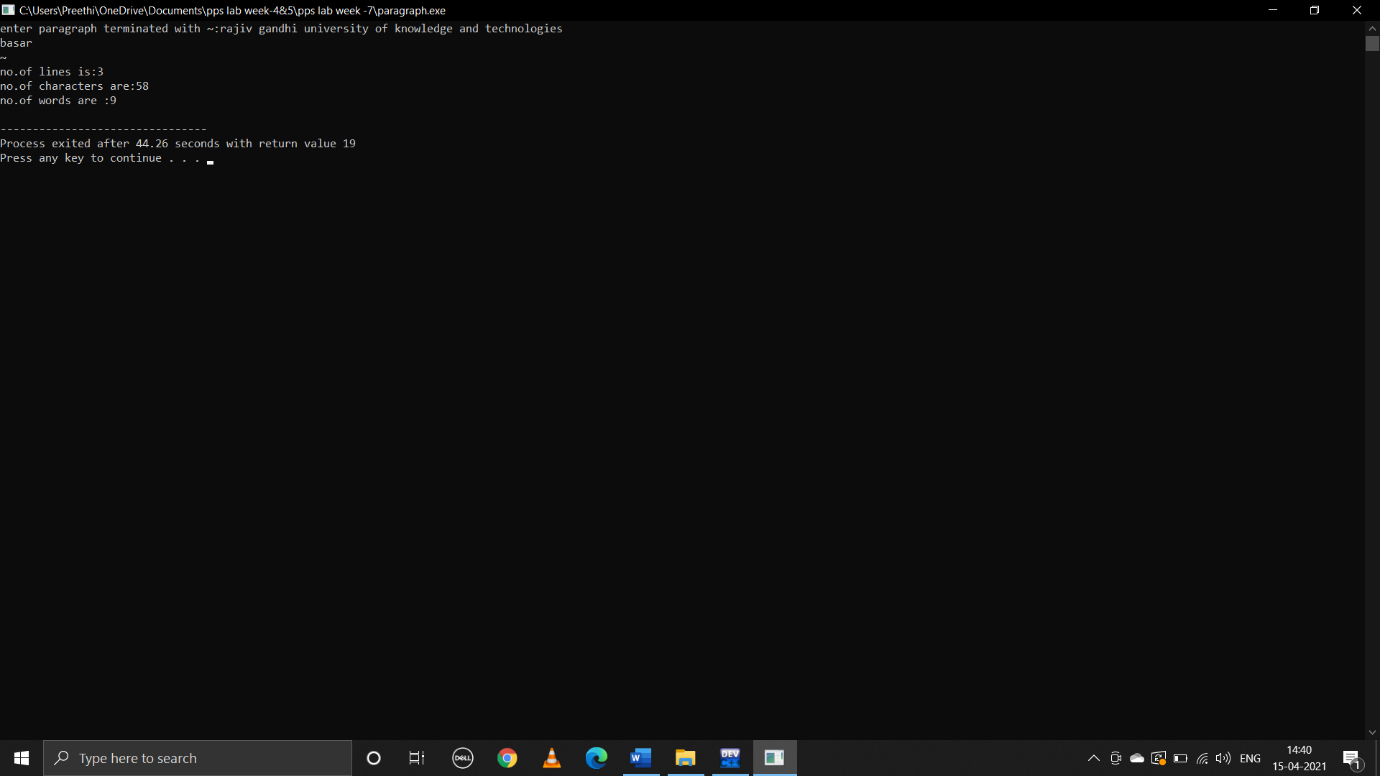
printf("no.of lines is:%d\n",line+1);

printf("no.of characters are:%d\n",ch);

printf("no.of words are :%d\n",word+1);

}

**Output:**

****

**WEEK-8**

**1.write a C function for the following tasks**

**i)calculating factorial**

**c program:**

#include<stdio.h>

void fact(int n);

main()

{

int x;

printf("enter x value:");

scanf("%d",&x);

fact(x);

}

void fact(int n)

{

int i,fact=1;

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

{

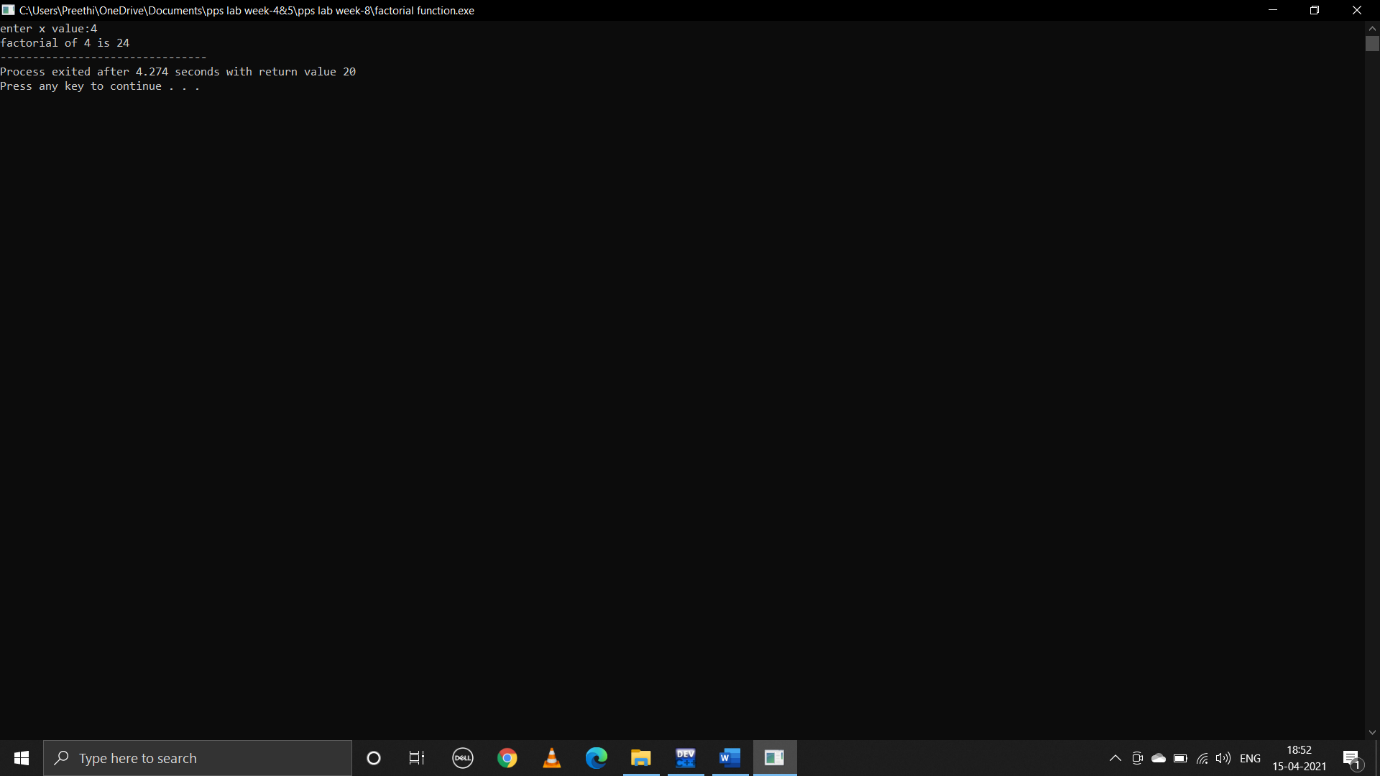
fact=fact\*i;

}

printf("factorial of %d is %d",n,fact);

}

**output:**

****

**ii).Find value of a given fibonacci term**

**c program:**

#include<stdio.h>

void main()

{

int x;

printf("enter x:");

scanf("%d",&x);

fib(x);

}

void fib(n)

{

int i,n1=0,n2=1,nth;

printf("%d\n%d\n",n1,n2);

for(i=0;i<n-2;i++)

{

nth=n1+n2;

printf("%d\n",nth);

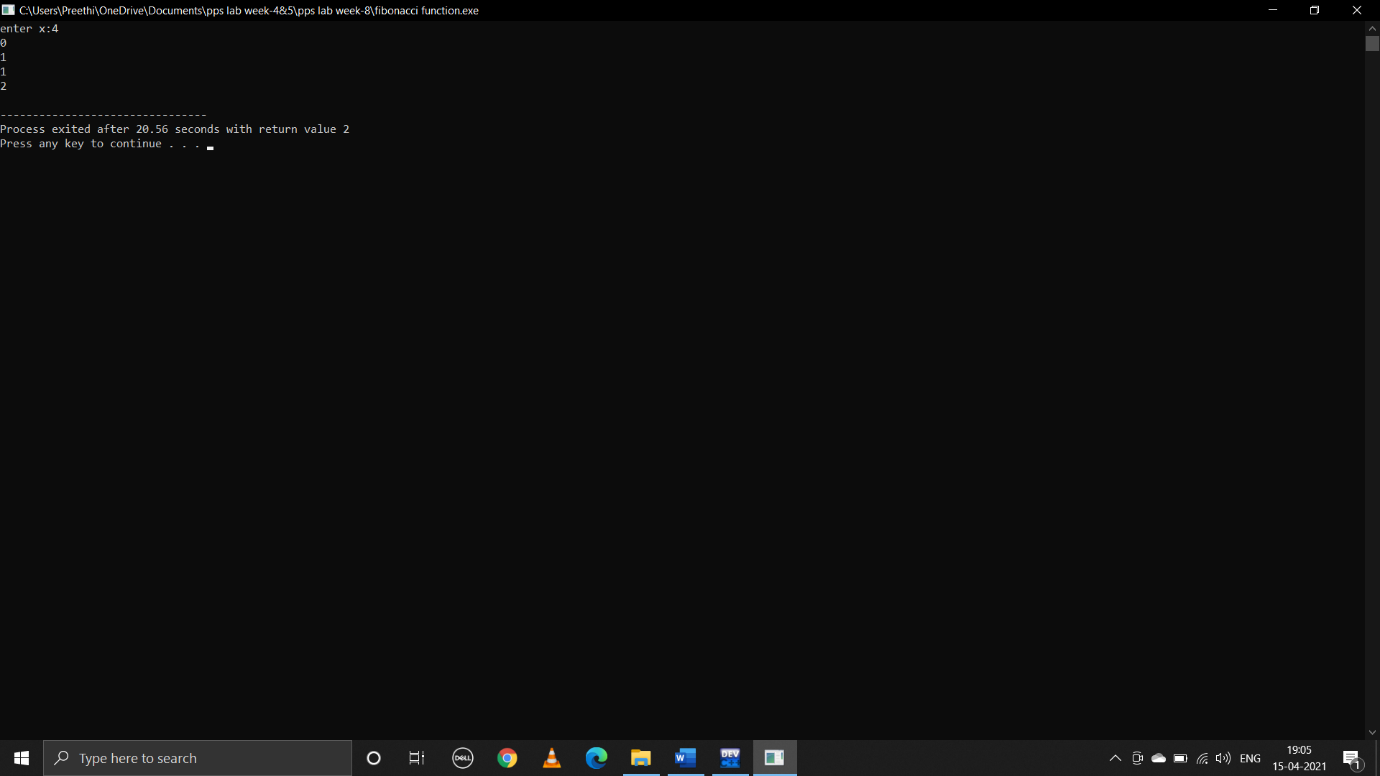
n1=n2;

n2=nth;

}

}

**output:**

****

**iii).swapping the values of two variable**

**c program:**

#include<stdio.h>

void swap(int,int);

main()

{

int a,b;

printf("enter a:");

scanf("%d",&a);

printf("enter b:");

scanf("%d",&b);

swap(a,b);

}

void swap(int x,int y)

{

printf("before swapping:\n a is %d\n b is %d\n",x,y);

int temp;

temp=x;

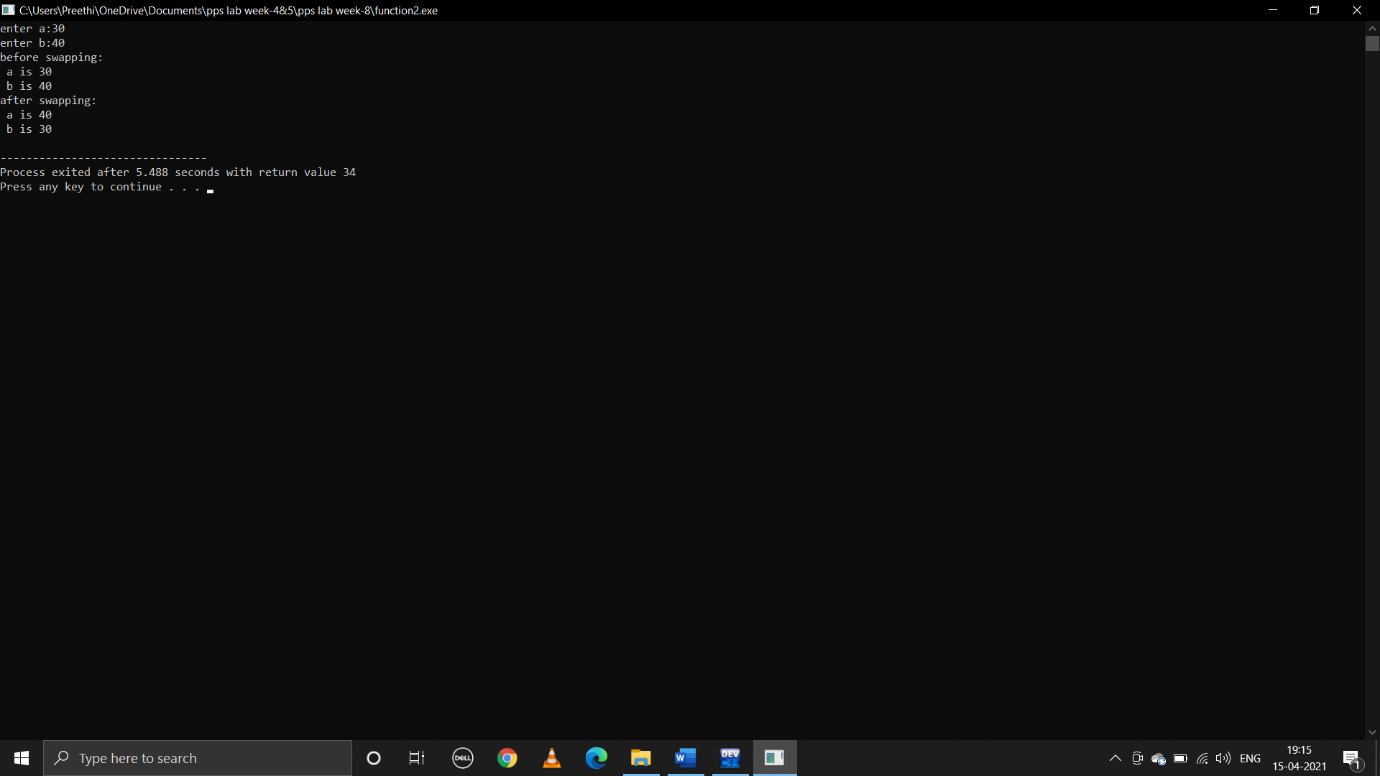
x=y;

y=temp;

printf("after swapping:\n a is %d\n b is %d\n",x,y);

}

**output:**

****

**2.write a c program that uses functions to perform the following operations**

**i) To insert a sub-string in to a given main string from a given postion**

**c program:**

#include<stdio.h>

#include<string.h>

void insert(char mainstr[100],char substr[100],int p);

main()

{

char mainstr[100];

char substr[100];

int p;

printf("enter main string:");

gets(mainstr);

printf("enter substring:");

gets(substr);

printf("enter position where to insert substring:");

scanf("%d",&p);

insert(mainstr,substr,p);

}

void insert(char mainstr[100],char substr[100],int p)

{

int i,m,n;

m=strlen(mainstr);

n=strlen(substr);

for(i=m-1;i>=p;i--)

{

mainstr[i+n]=mainstr[i];

}

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

{

mainstr[i+p]=substr[i];

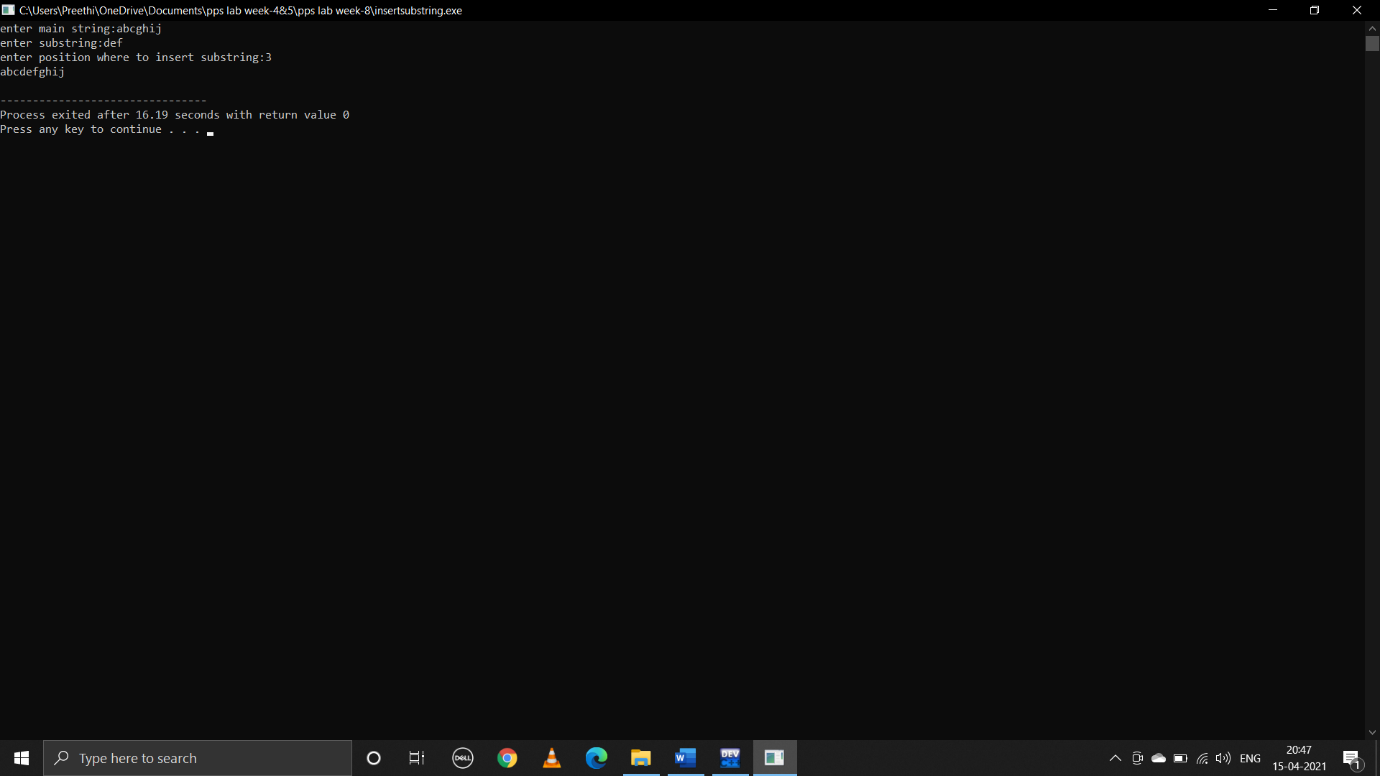
}

mainstr[m+n]=’\0’;

puts(mainstr);

}

**output:**

****

**ii) To delete n characters from a given position in a given string**

**c program:**

#include<stdio.h>

#include<string.h>

char del(char s1[],int,int,int);

int main()

{

char s1[100];

int loc,n,len;

printf("enter string:\n");

gets(s1);

printf("enter location:\n");

scanf("%d",&loc);

printf("enter no.of characters to delete:\n");

scanf("%d",&n);

len=strlen(s1);

del(s1,loc,n,len);

}

char del(char s1[],int loc,int n,int len)

{

int i,j;

if(loc>len)

{

printf("invalid location\n");

}

else

{

char s2[100];

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

{

s2[i]=s1[i];

}

for(j=loc+n;j<len;j++,i++)

{

s2[i]=s1[j];

}

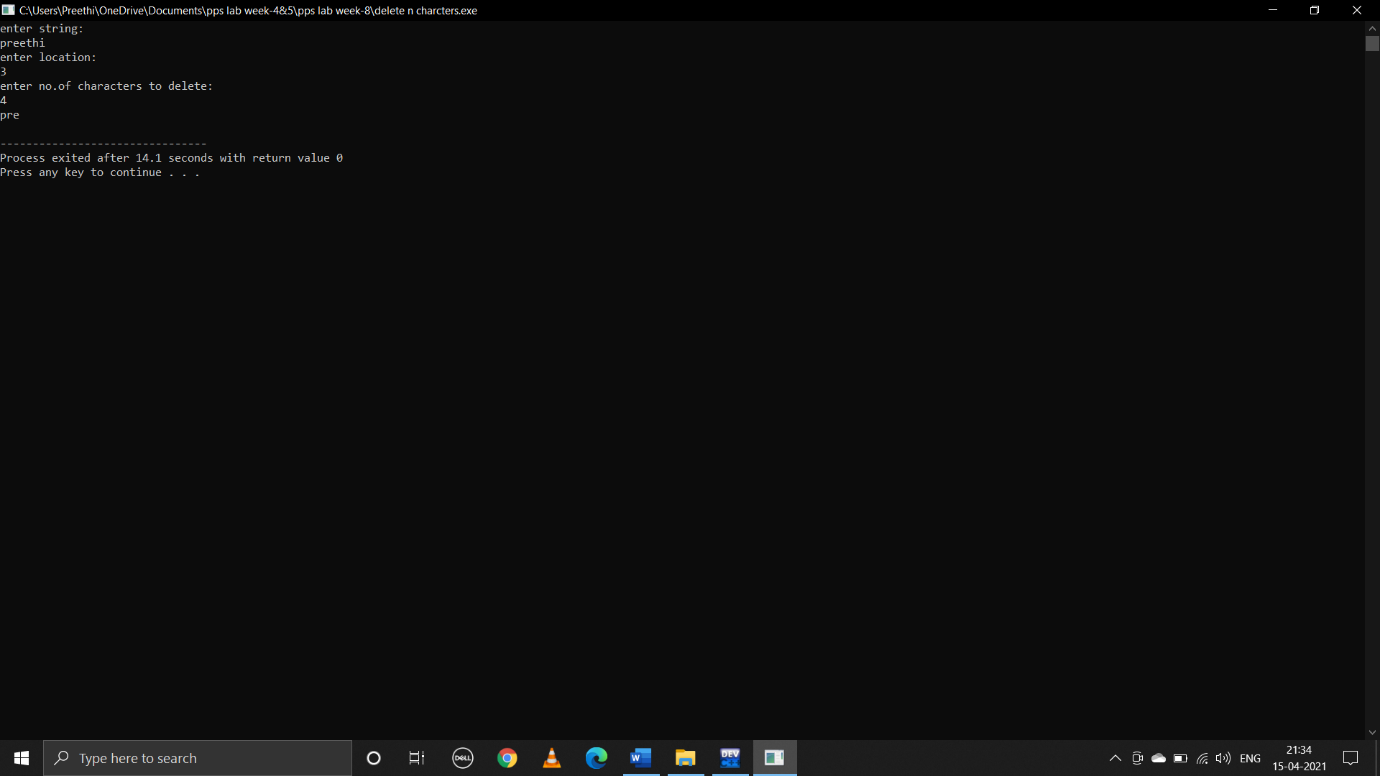
s2[i]='\0';

puts(s2);

}

}

**output:**

****

**WEEK-9**

**1.write the following recursive c function**

**c program:**

#include<stdio.h>

int fact(int n);

int main()

{

int x;

printf("enter number:");

scanf("%d",&x);

fact(x);

printf("factorial of number is:%d",fact(x));

return 0;

}

int fact(int n)

{

if(n>=1)

{

return n\*fact(n-1);

}

else

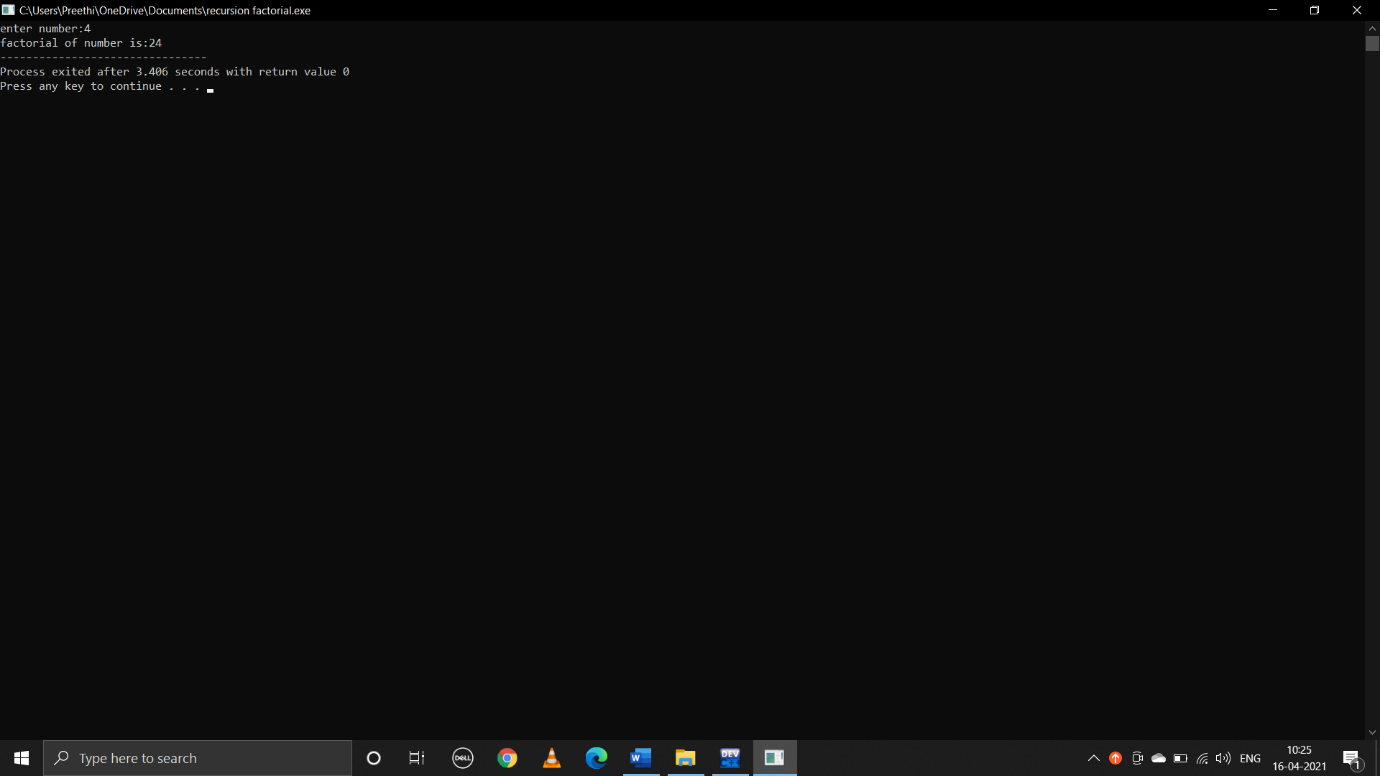
{

return 1;

}

}

**output:**

****

**ii)Nth fibonacci number**

**c program:**

#include<stdio.h>

int fib(int n);

main()

{

int x,i;

printf("enter no.of terms:");

scanf("%d",&x);

printf("fibonacci series is:\n");

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

{

printf("%d\n",fib(i));

}

return 0;

}

int fib(int n)

{

if(n==0)

{

return 0;

}

else if(n==1)

{

return 1;

}

else

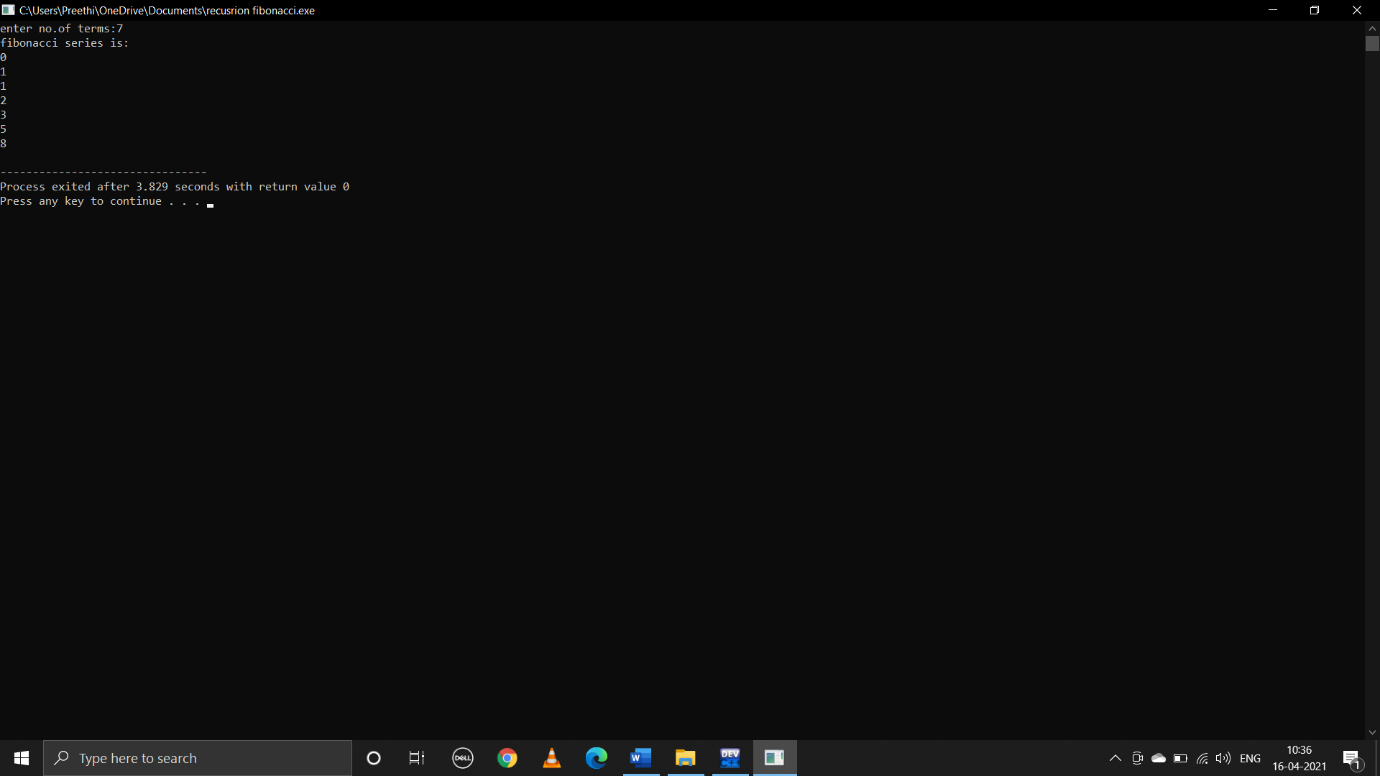
{

return(fib(n-1)+fib(n-2));

}

}

**output:**

****

**iii)Reverse of a given string**

**c program:**

#include<stdio.h>

void reverse(char \*str);

main()

{

char s[100];

printf("enter string:");

gets(s);

puts(s);

reverse(s);

return 0;

}

void reverse(char\*str)

{

if((\*str)!='\0')

{

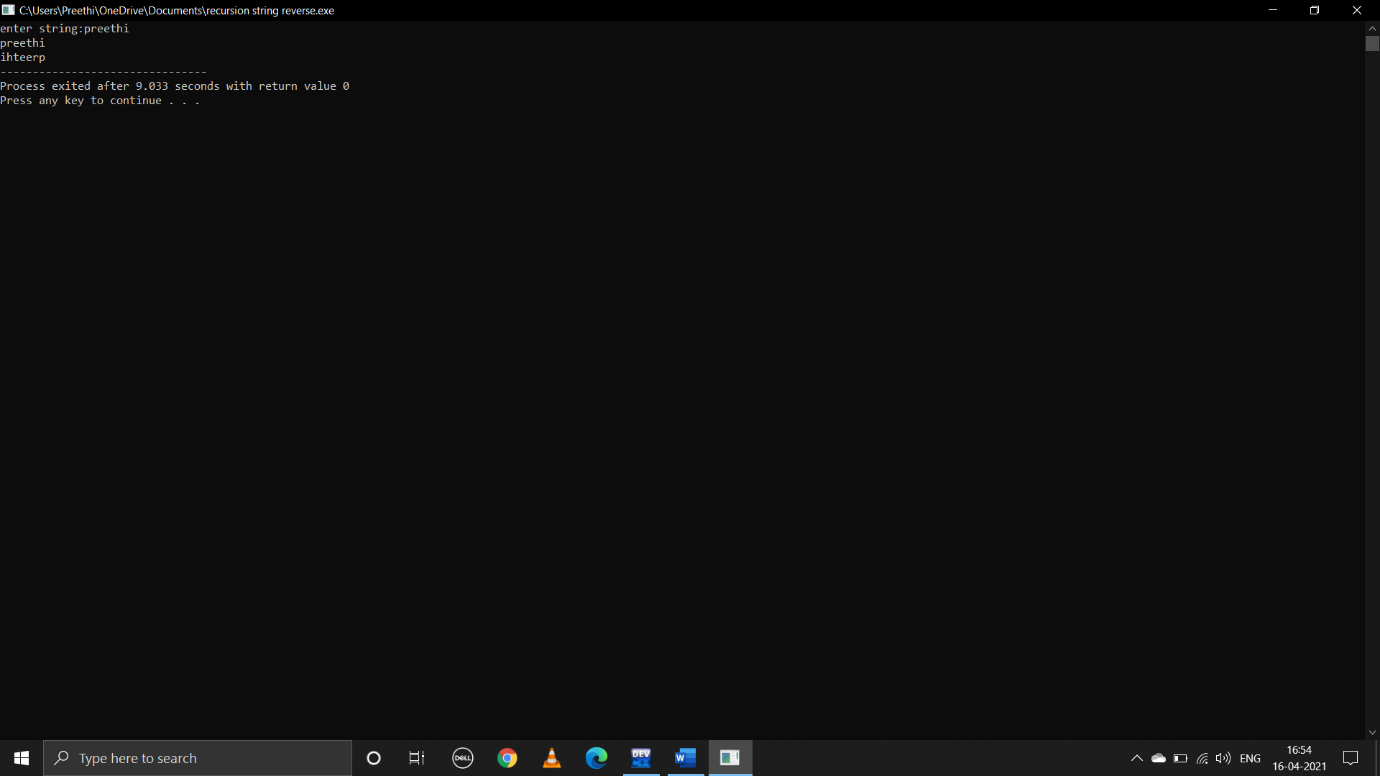
reverse(str+1);

printf("%c",\*str);

}

}

**output:**

****

**iv)Reverse of a given number**

**c program:**

#include<stdio.h>

main()

{

int x;

printf("enter x value:");

scanf("%d",&x);

rev(x);

return 0;

}

void rev(int a)

{

int rem;

if(a==0)

{

return;

}

else

{

rem=a%10;

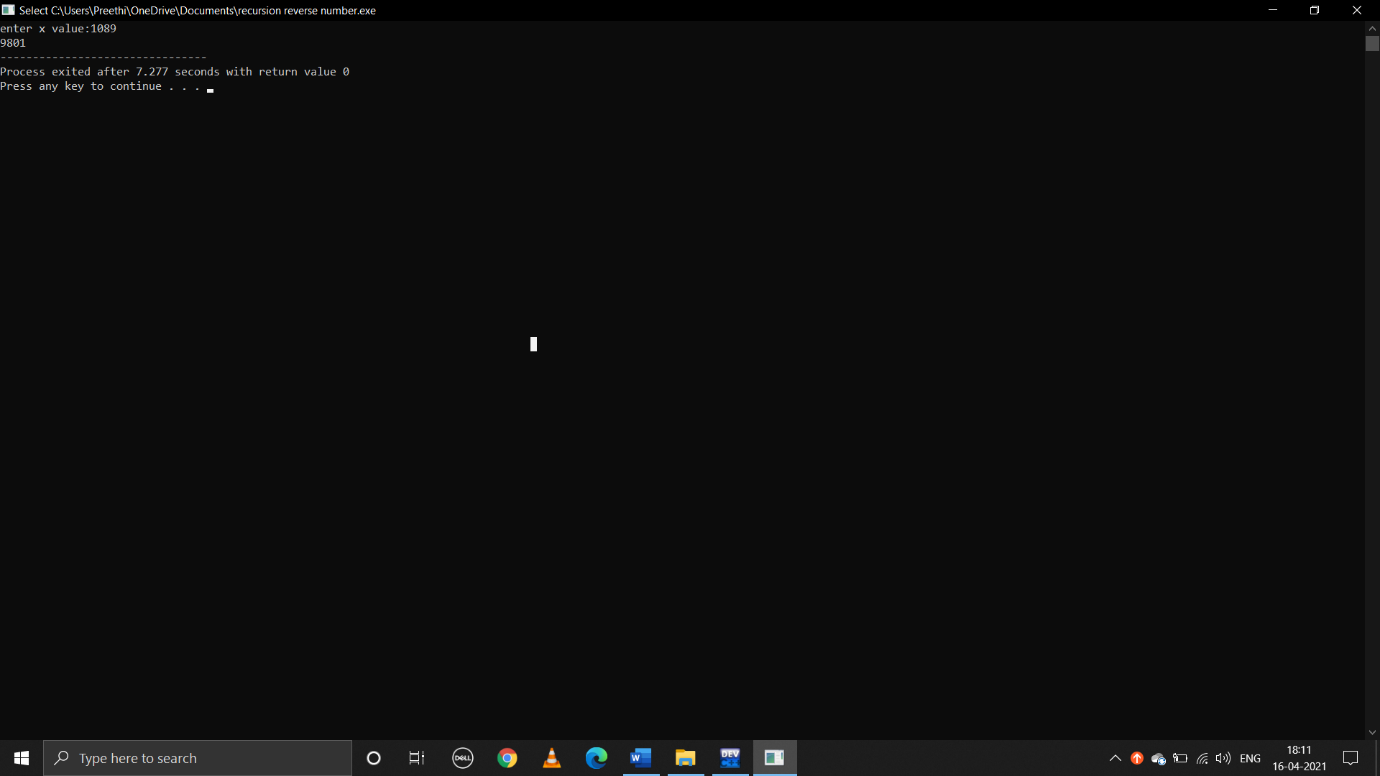
printf("%d",rem);

rev(a/10);

}

}

**output:**

****

**WEEK-8**

**1.write a C program to maintain a record of “n” student details using an array of structures with four fields (roll number,name,marks and grade).assume appropriate data type for each field.print the marks of the student,given the student rollnumber as input.**

**C program:**

#include<stdio.h>

struct student

{

int rollnumber;

char name[20];

int marks;

char grade[10];

};

void main()

{

int n,i;

printf("enter number of students:");

scanf("%d",&n);

struct student std[n];

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

{

printf("enter roll number:");

scanf("%d",&std[i].rollnumber);

printf("enter name:");

scanf("%s",std[i].name);

printf("enter marks:");

scanf("%d",&std[i].marks);

printf("enter grade:");

scanf("%s",std[i].grade);

}

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

{

printf("%d\n%s\n%d\n%s\n",std[i].rollnumber,std[i].name,std[i].marks,std[i].grade);

}

int r,j;

printf("enter rollnumber:");

scanf("%d",&r);

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

{

if(std[j].rollnumber==r)

{

printf("%d",std[j].marks);

break;

}

}

if(j==n)

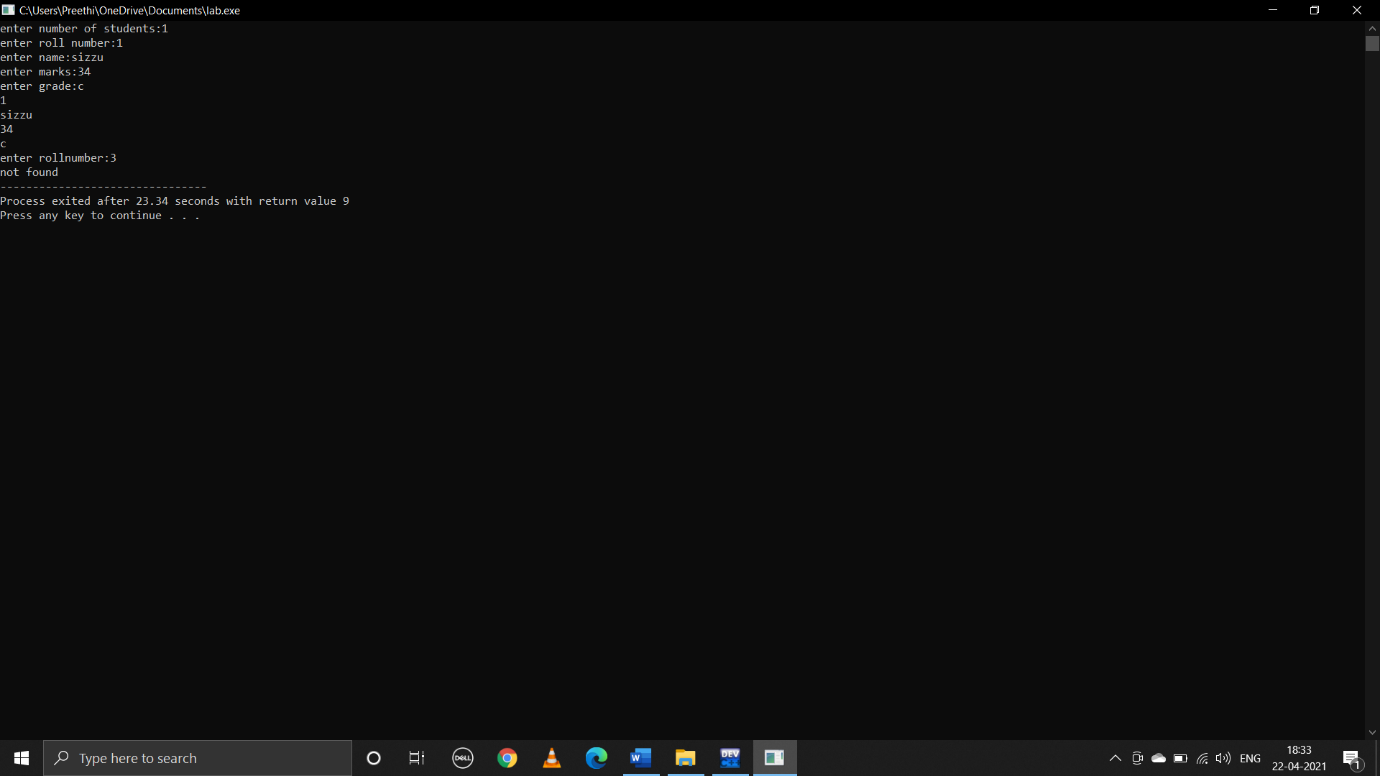
{

printf("not found");

}

}

**Output:**

****

**2.Define structure called cricket that will describe the information player name,team name,batting average.using cricket,declare an array player information about all 10 players and print team wise list containing names of the player with their batting average**

**C program:**

#include<stdio.h>

struct cricket

{

char name[20];

char teamname[20];

float battingaverage;

};

void main()

{

int i,j,n;

printf("enter no.of players:");

scanf("%d",&n);

struct cricket a[n];

struct cricket t;

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

{

printf("enter name:\n");

scanf("%s",a[i].name);

printf("enter team name:\n");

scanf("%s",a[i].teamname);

printf("enter batting average:\n");

scanf("%f",&a[i].battingaverage);

}

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

{

printf("%s\t%s\t%f\t",a[i].name,a[i].teamname,a[i].battingaverage);

}

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

{

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

{

if(strcmp(a[i].teamname,a[j].teamname)>0)

{

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

}

j=0;

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

{

if(strcmp(a[i].teamname,a[j].teamname)!=0||i==0)

{

printf("\n team name:%s",a[i].teamname);

j=i;

}

printf("\n player name=%s",a[i].name);

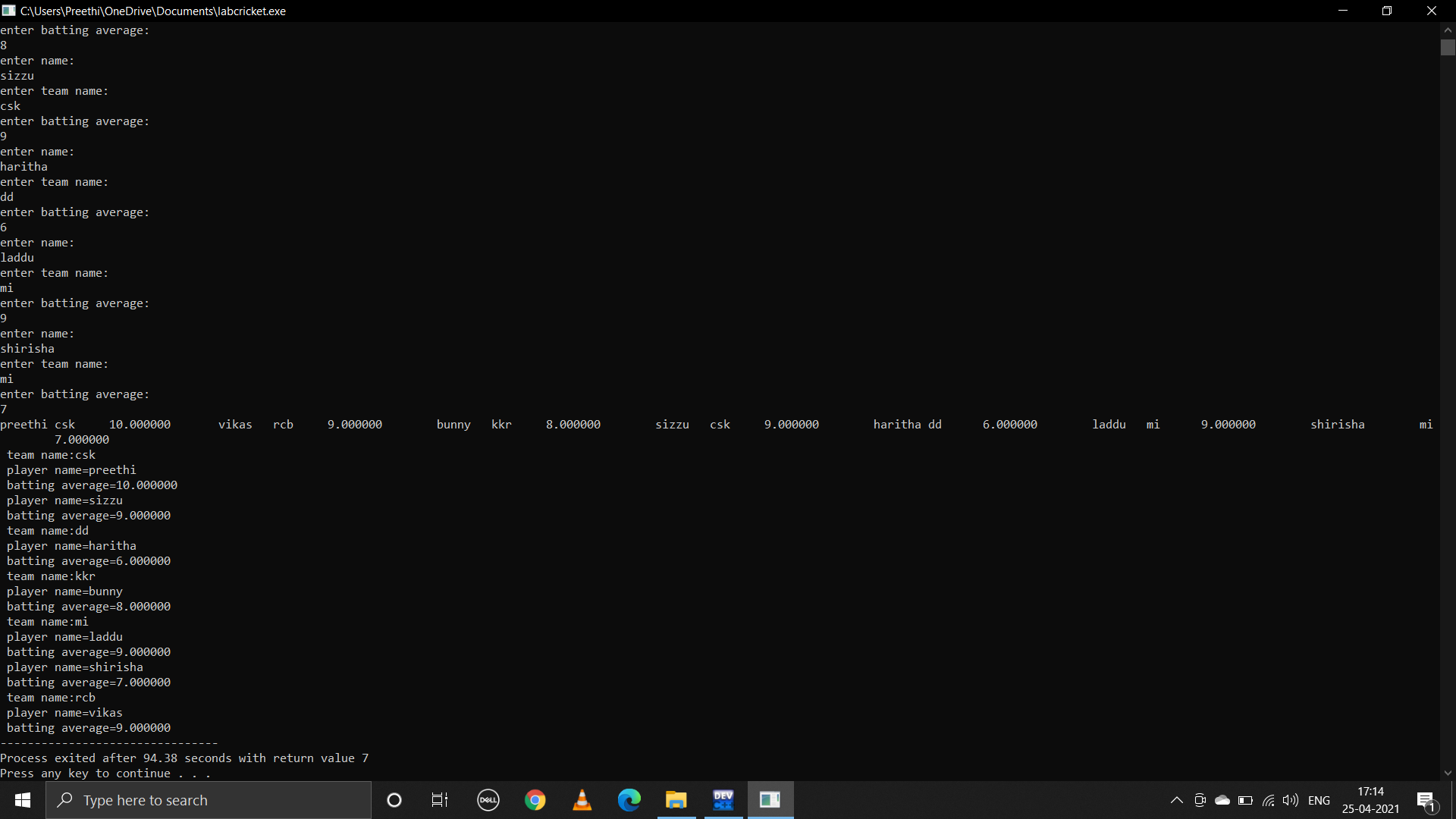
printf("\n batting average=%f",a[i].battingaverage);

}

return 0;

}

**Output:**

****

**3.write a program using pointers to compute the sum of all elements in an array**

**C program:**

#include<stdio.h>

void main()

{

int n,i;

printf("enter size of an array:");

scanf("%d",&n);

int a[n];

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

{

printf("enter %d element:",i);

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

}

int \*p;

p=&a[0];

int sum=0;

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

{

printf("%d ",\*(p+i));

sum=sum+\*(p+i);

}

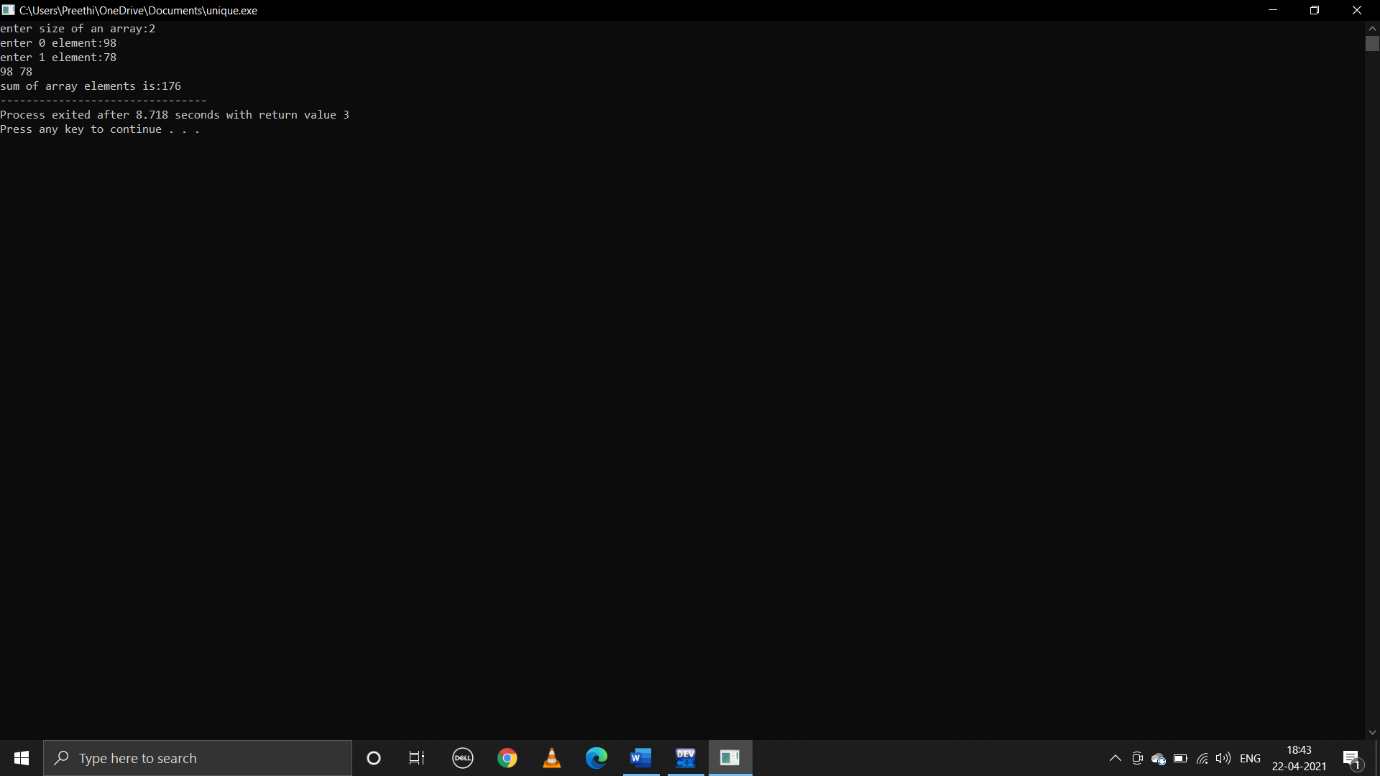
printf("\n");

printf("sum of array elements is:");

printf("%d",sum);

}

**Output:**

****

**4.write a program to print the elements of a structure using pointers**

**C program:**

#include<stdio.h>

struct details

{

char name[20];

int age;

char city[20];

};

void main()

{

int n,i;

printf("enter n:");

scanf("%d",&n);

struct details a[n];

struct details\*p;

p=&a[0];

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

{

printf("enter name:\n");

scanf("%s",(p+i)->name);

printf("enter age:\n");

scanf("%d",&(p+i)->age);

printf("enter city:\n");

scanf("%s",(p+i)->city);

}

p=&a[0];

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

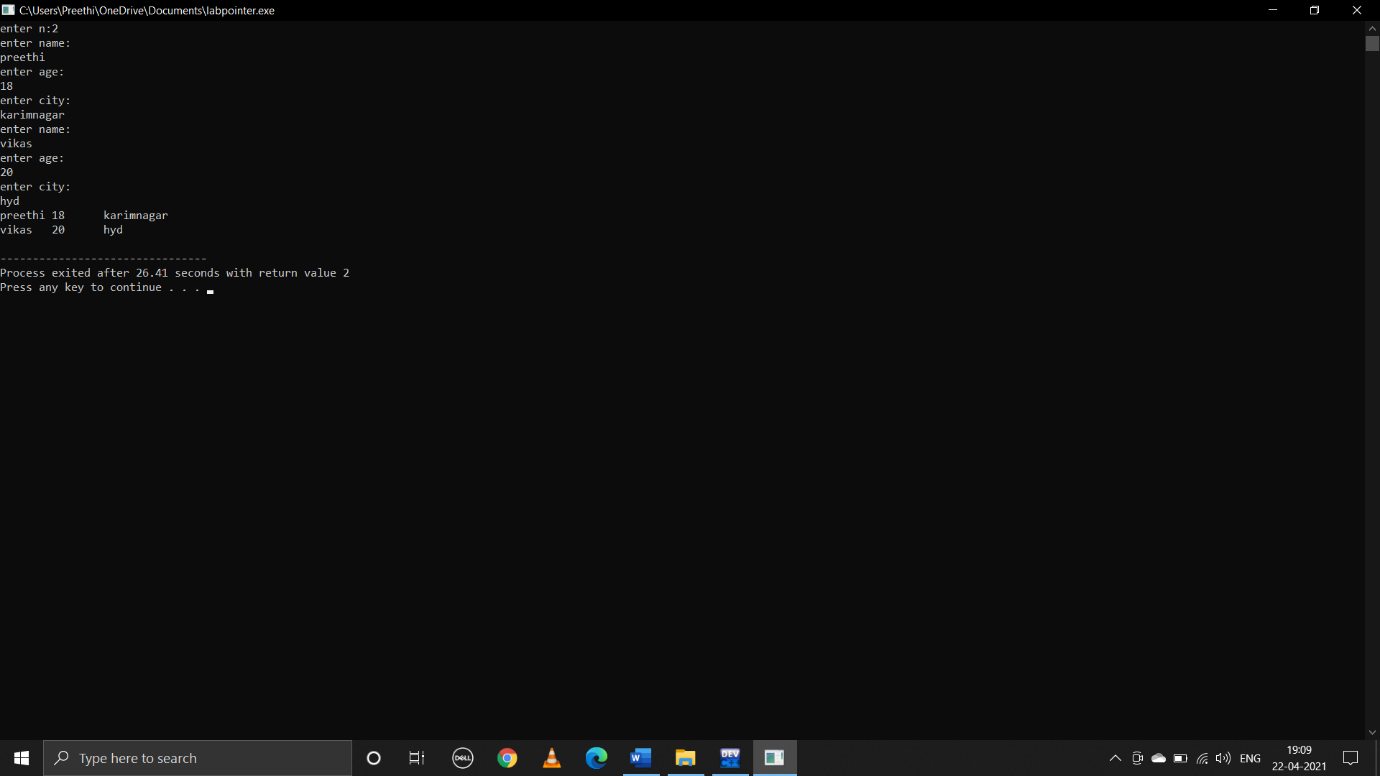
{

printf("%s\t%d\t%s\t\n",(p+i)->name,(p+i)->age,(p+i)->city);

}

}

**Output:**

****

**WEEK-9**

**1.write a C program that creats an employee text file?Records are empid,empname,designation,qualification,salary,experience,research work,address,city,phone?**

**C program:**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE\*pointer;

int empid,salary,adress,experience;

char empname[100];

char designation[100];

char qualification[100];

char researchwork[100];

long phone;

char city[100];

pointer=fopen("username.txt","w");

if(pointer==NULL)

{

printf("file doesn't exist\n");

return 0;

}

printf("enter employee ID:");

scanf("%d",&empid);

fprintf(pointer,"\nemployee ID:%d",empid);

printf("enter employee name:");

scanf("%s",empname);

fprintf(pointer,"\nemployee name:%s",empname);

printf("enter designation:");

scanf("%s",designation);

fprintf(pointer,"\ndesignation:%s",designation);

printf("enter qualification:");

scanf("%s",qualification);

fprintf(pointer,"\nqualification:%s",qualification);

printf("enter salary:");

scanf("%d",&salary);

fprintf(pointer,"\nsalary:%d",salary);

printf("enter experience:");

scanf("%d",&experience);

fprintf(pointer,"\nexperince:%d",experience);

printf("enter researchwork:");

scanf("%s",researchwork);

fprintf(pointer,"\nresearchwork:%s",researchwork);

printf("enter adress:");

scanf("%d",&adress);

fprintf(pointer,"\nadress:%d",adress);

printf("enter city:");

scanf("%s",city);

fprintf(pointer,"\ncity:%s",city);

printf("enter phone:");

scanf("%d",phone);

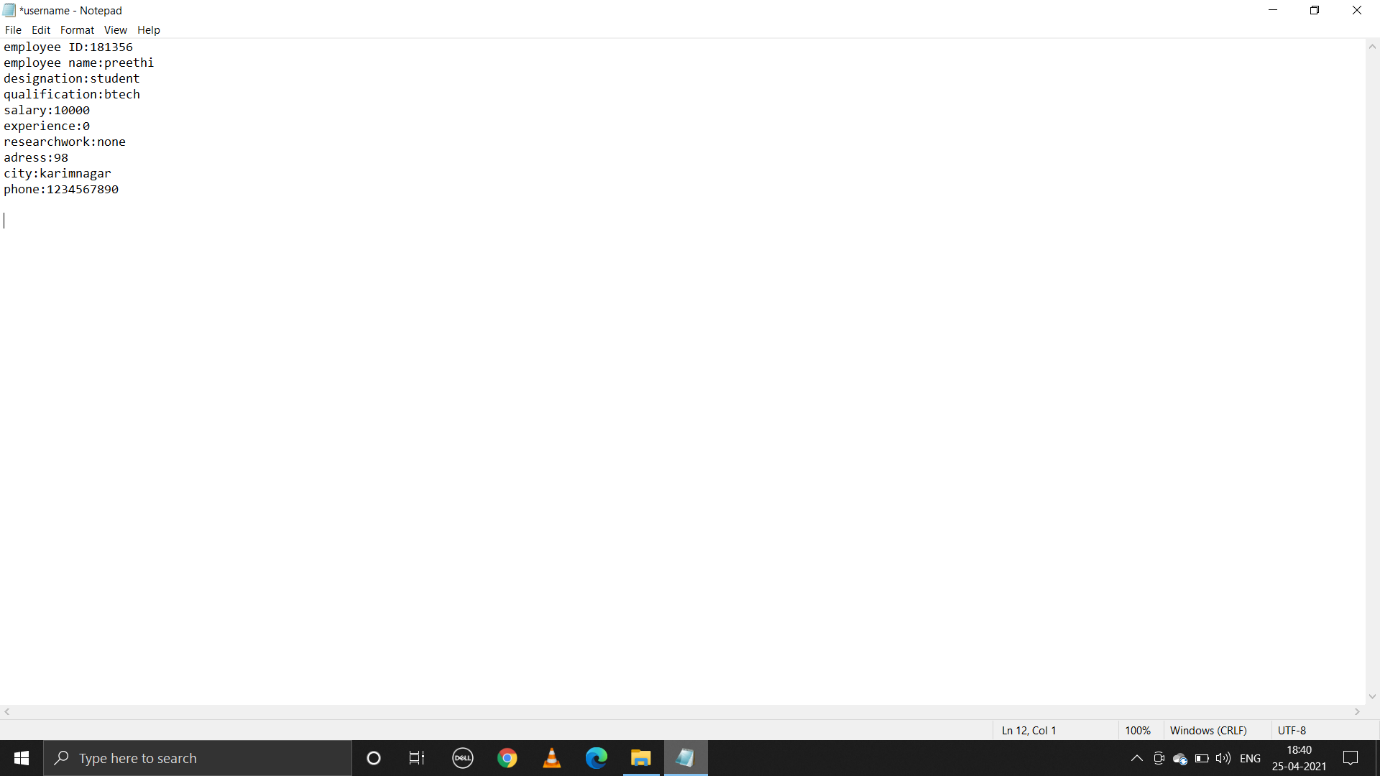
fprintf(pointer,"\nphone:%d",phone);

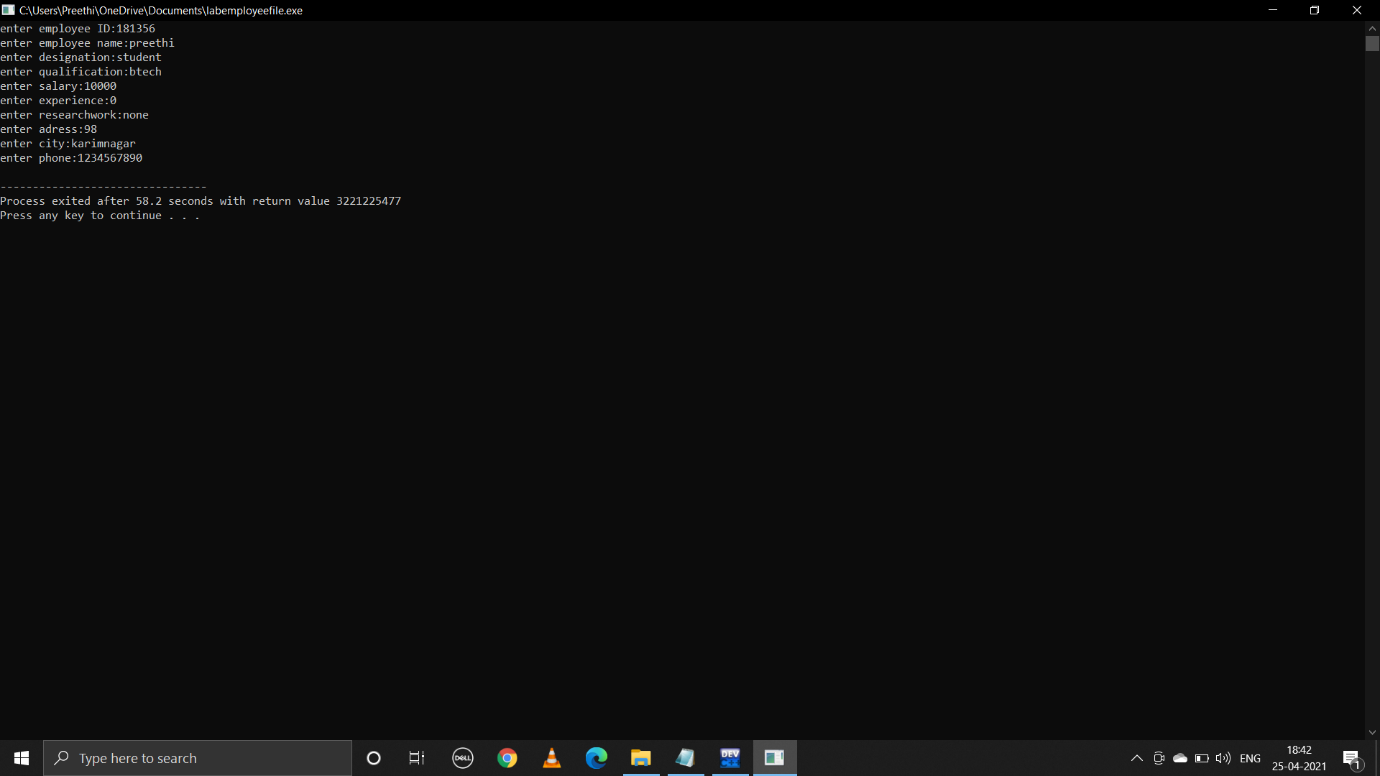
fclose(pointer);

return 0;

}

**Output:**

****

****

**2.write a C program that manipulates the above text file.The program must implements the operation to modify a record,delete a record,delete a record**

**and append new record**

**C program:**

#include<stdio.h>

#include<stdio.h>

int main()

{

FILE\*fpointer;

char hobbies[100];

char strengths[100];

fpointer=fopen("username.txt","r++");

fpointer=fopen("username.txt","a");

if(fpointer==NULL)

{

printf("file does not exists\n");

return 0;

}

printf("enter your hobbies:");

scanf("%s",hobbies);

fprintf(fpointer,"\nhobbies:%s",hobbies);

printf("enter your strengths:");

scanf("%s",strengths);

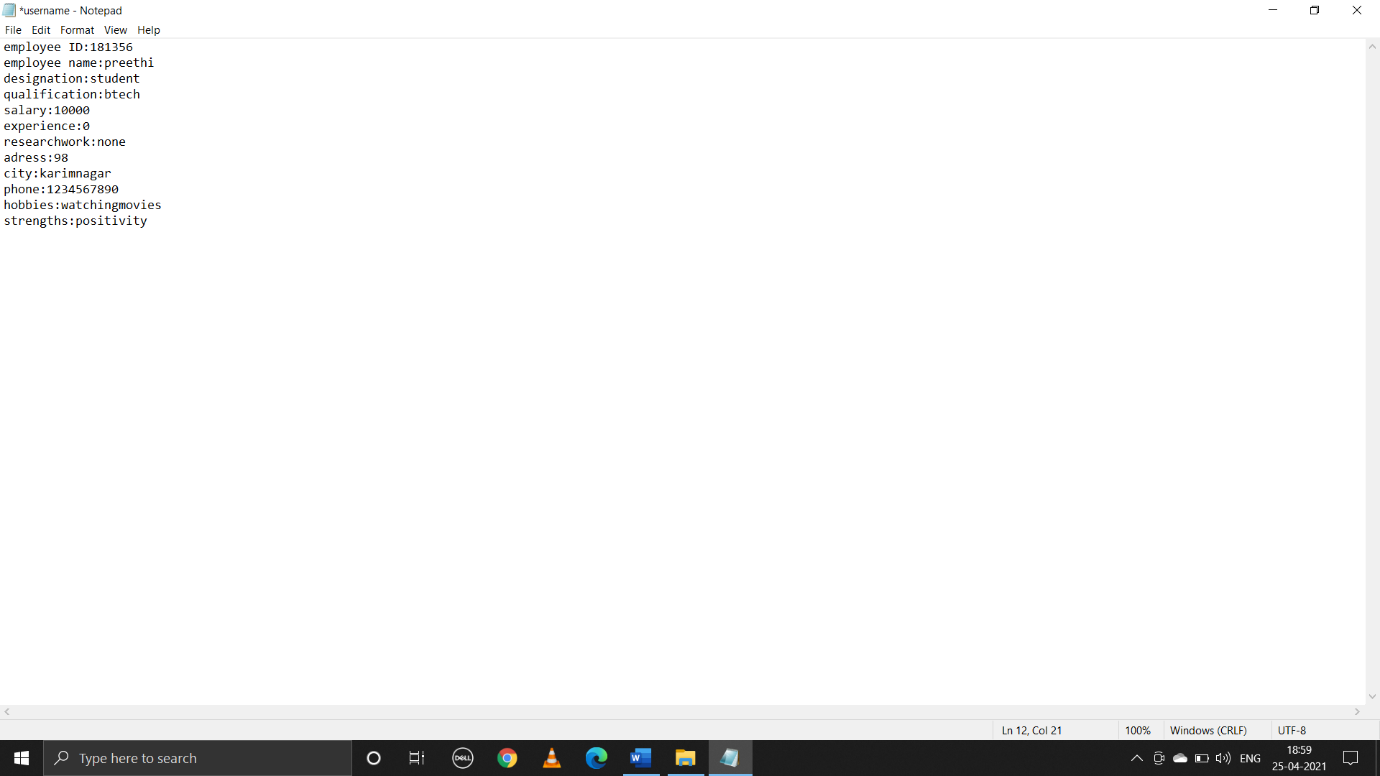
fprintf(fpointer,"\nstrengths:%s",strengths);

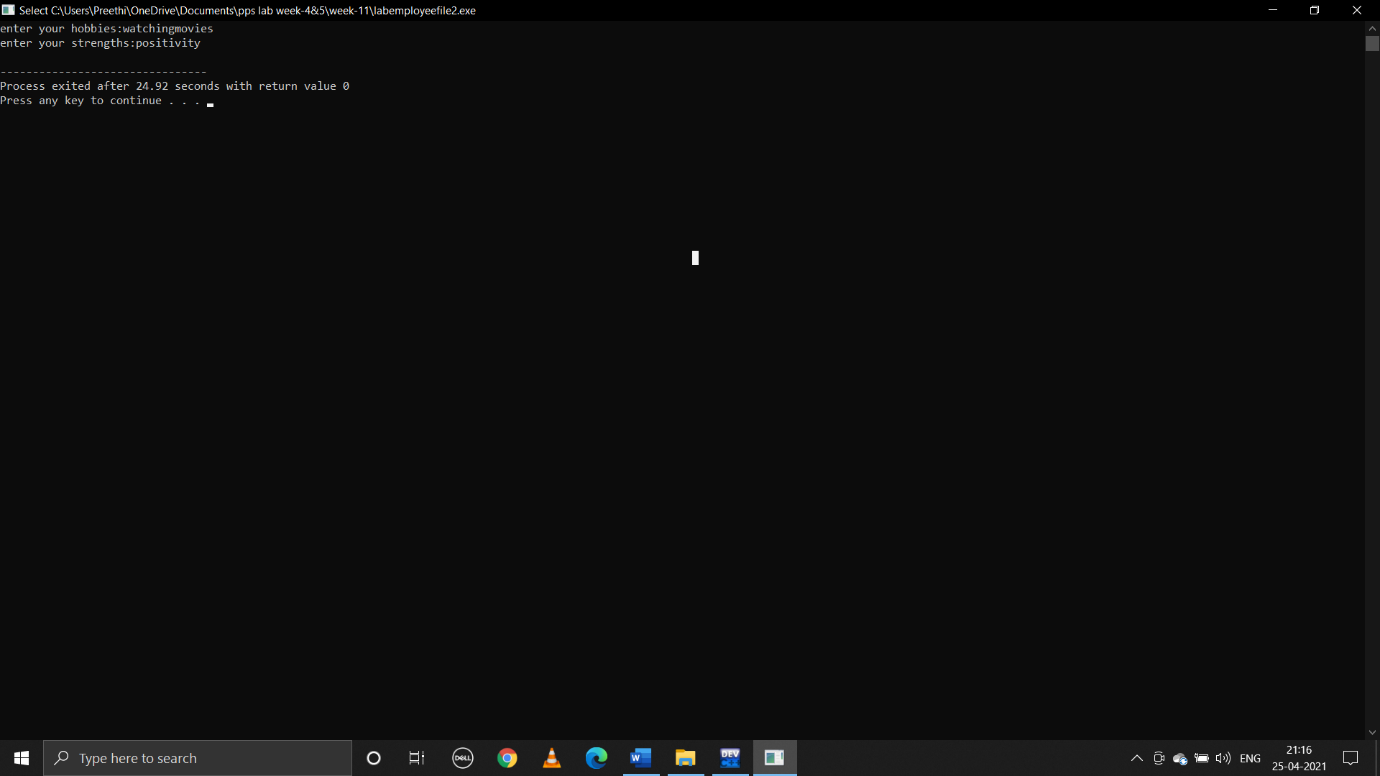
remove(strengths);

return 0;

}

**Output:**

****

****