Design and Analysis of Algorithms Lab

Submitted By;

T.M.Dheenadayalan

9918004024

**AIM:**

To find the GCD by implementing Euclids Algorithm

**ALGORITHM:**

1.Enter any two integers.

2.Check whether it is a non zero number or not.

3.FInd the modulus and store it in r.

4.Check whether m,n,r are equal.

5.If it is equal print the GCD as m. while n != 0

r = m mod n m = n

n = r return m **PROGRAM:**

#include <iostream> using namespace std; void main()

{

int m, n;

cout<<"Enter two integer numbers:";

cin>>m>>n;

while (n > 0)

{

int r = m % n;

m = n;

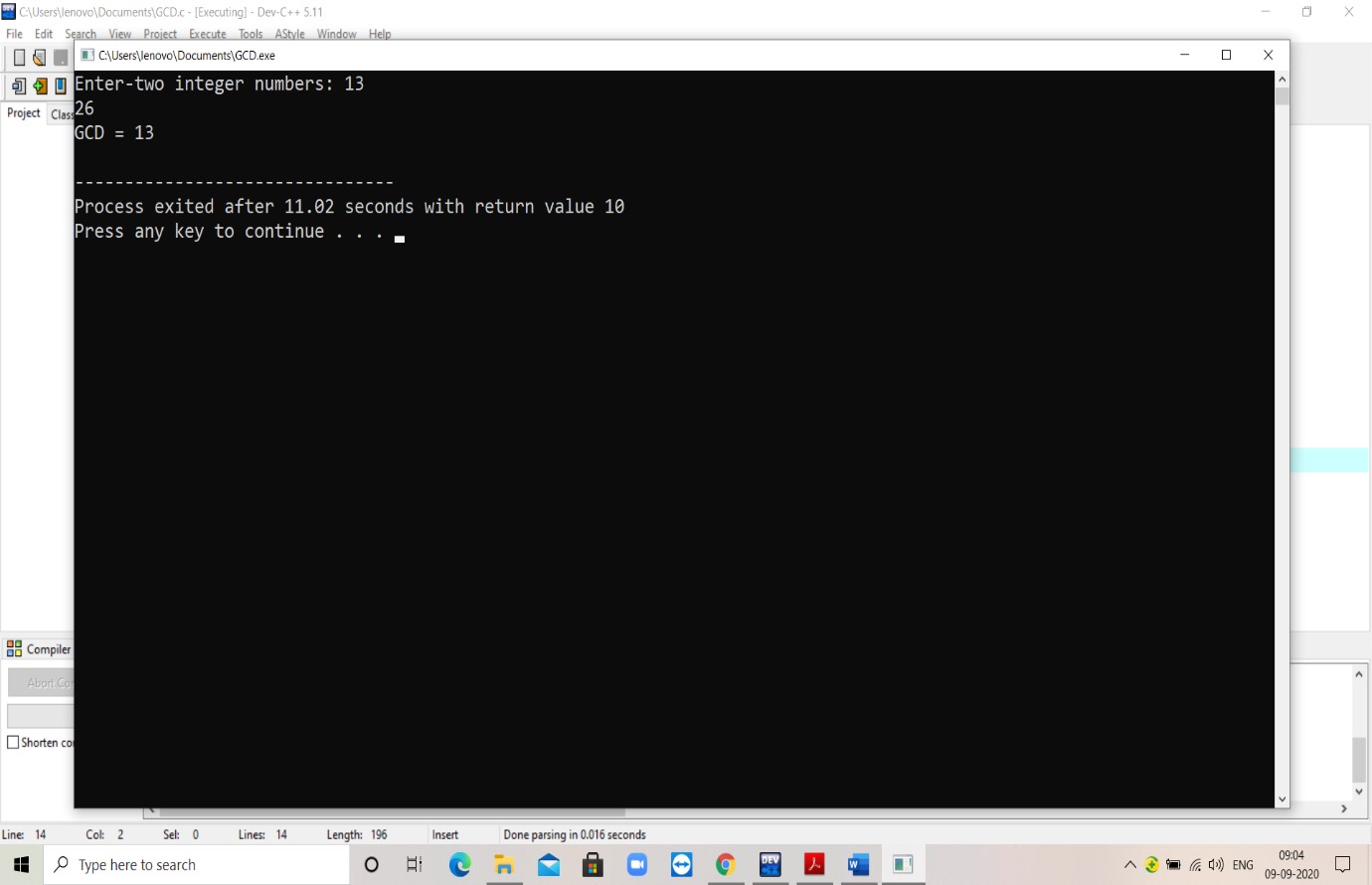
n = r;

}

cout<<"GCD ="<<” “<<m;

}

**OUTPUT:**



**AIM:**

To write the code for finding GCD using consecutive checking

algorithm

**ALGORITHM:**

1.Assign value of sml{m, n} to n

2 .Divide m by t.

If remainder is 0, go to Step 3; Otherwise, go to Step 4

3. Divide n by t.

If remainder is 0, return t and stop; Otherwise, go to Step 4

4. Decrease t by 1 and go to Step 2

5.Print the GCD

**PROGRAM:**

#include<iostream> using namespace std; int gcd(int m,int n){ int t,sml;

if(m<n){

sml=m;

} else{ sml=n;

} t=sml; while(t>0){

if(m%t==0 && n%t==0){

return t;

}

t=t-1;

}

}

int main(){

int m,n,res;

cout<<"Enter Two Values";

cin>>m>>n;

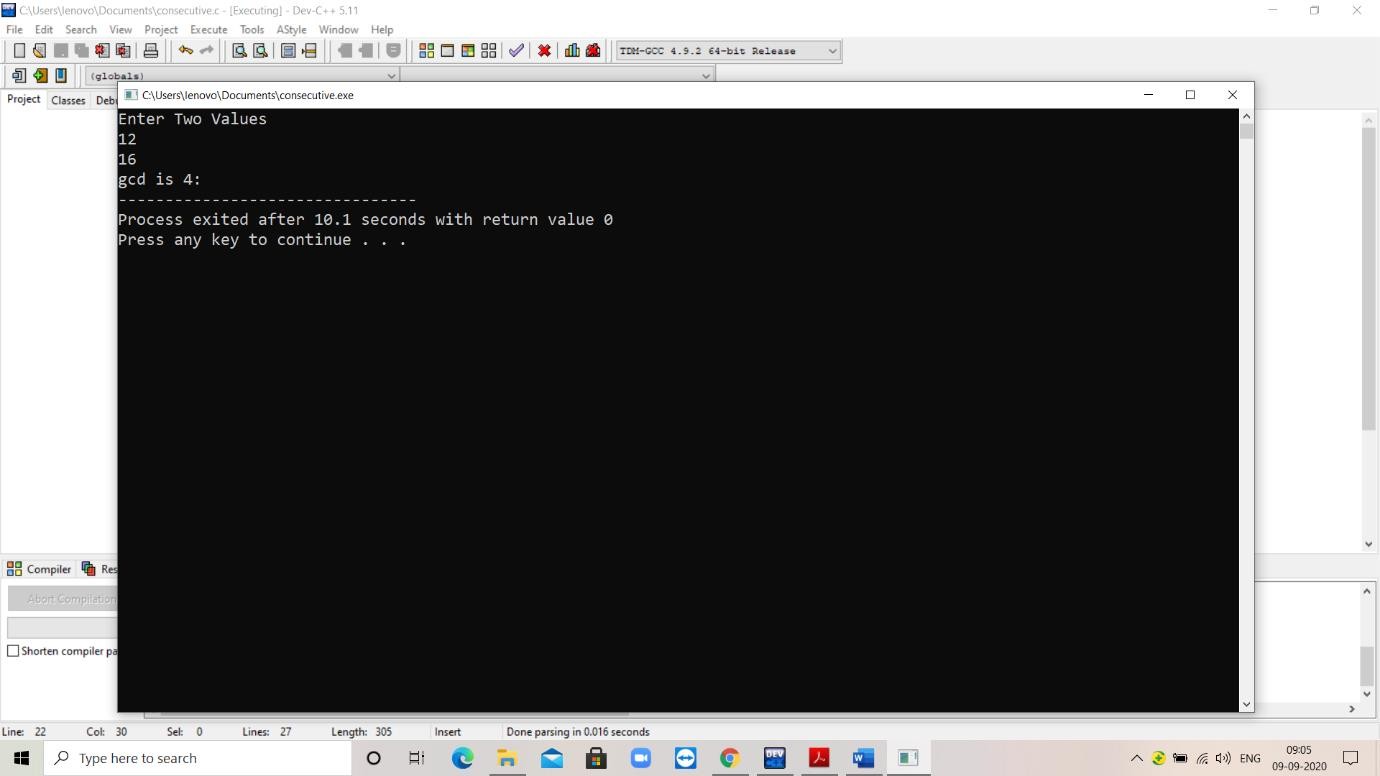
res=gcd(m,n);

cout<<"gcd is "<<” “<<res;

return 0;

}

**OUTPUT:**



**AIM:**

To find the GCD using middle school procedure algorithm.

**ALGORITHM:**

Find the prime factorization of m. Find the prime factorization of n. Find all the common prime factors.

Compute the product of all the common prime factors and return it as gcd(m, n).

**PROGRAM:**

#include<iostream> using namespace std: int GCD (int a,int b)

{

int i,c,n=0,m=0,ans=1; c=a>=b?b:a; for(i=2;i<=c;i++)

{ n=0; m=0;

if(a%i==0){

n=1;

a=a/i;

} if(b%i==0){ m=1;

b=b/i;

} if(n==1&&m==1){ ans=ans\*i;

}

if(n==1|| m==1){

i--;

}

}

return(ans);

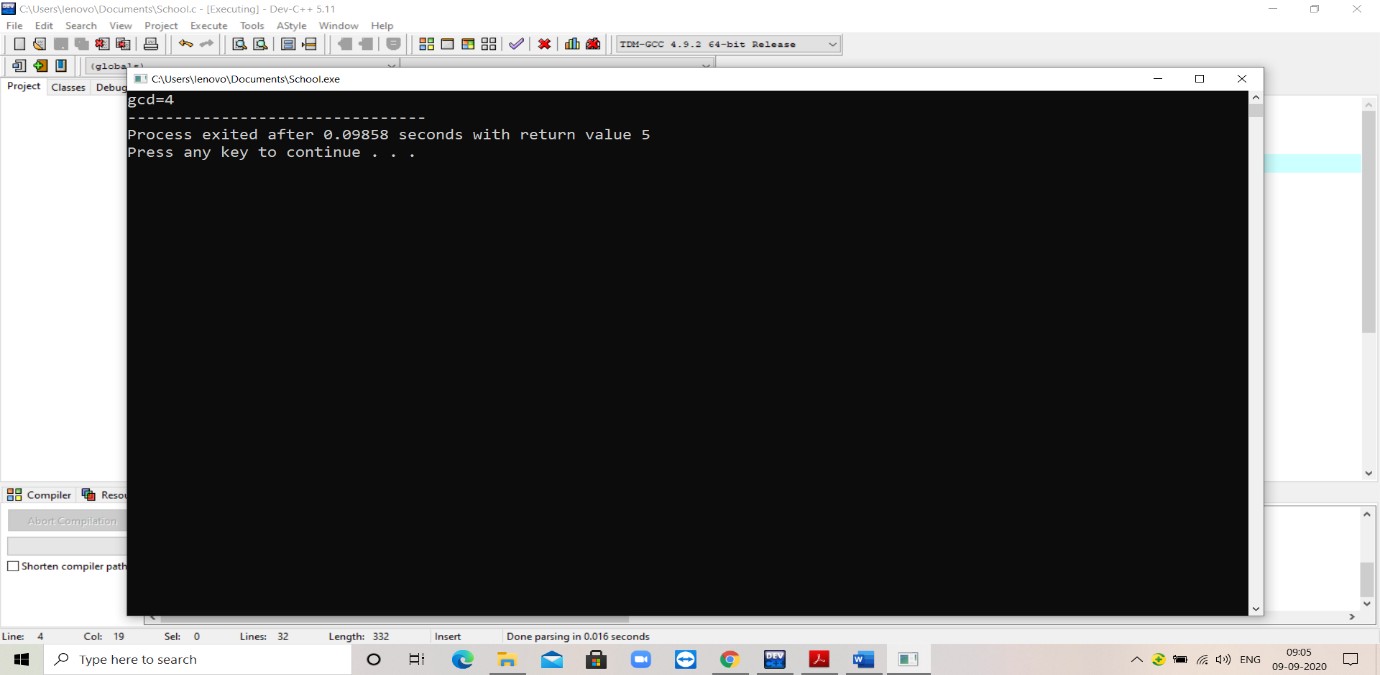
}

int main(){

int a=12,b=16,gcd; gcd=GCD(a,b); cout<<"gcd ="<<gcd;

}

**OUTPUT:**



**EXP NO:2**

**AIM:**

Write a program to find the maximum element in given array

**ALGORITHM:**

**1.**Create a function findmax

**2.**Put For loop check the given number is max or not

**3.**If given number is max return max element

**4.**Create main function

**5.**Give some array elements

**6.**Print Max of given number is findmax(a,n)

**PROGRAM:**

#include<iostream>

using namespace std;

int findmax(int a[],int n)

{

int maxelement=a[0];

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

{

if(maxelement<a[i])

{

maxelement=a[i];

}

}

return maxelement;

}

int main()

{

int a[]={11,25,17,96,58,336};

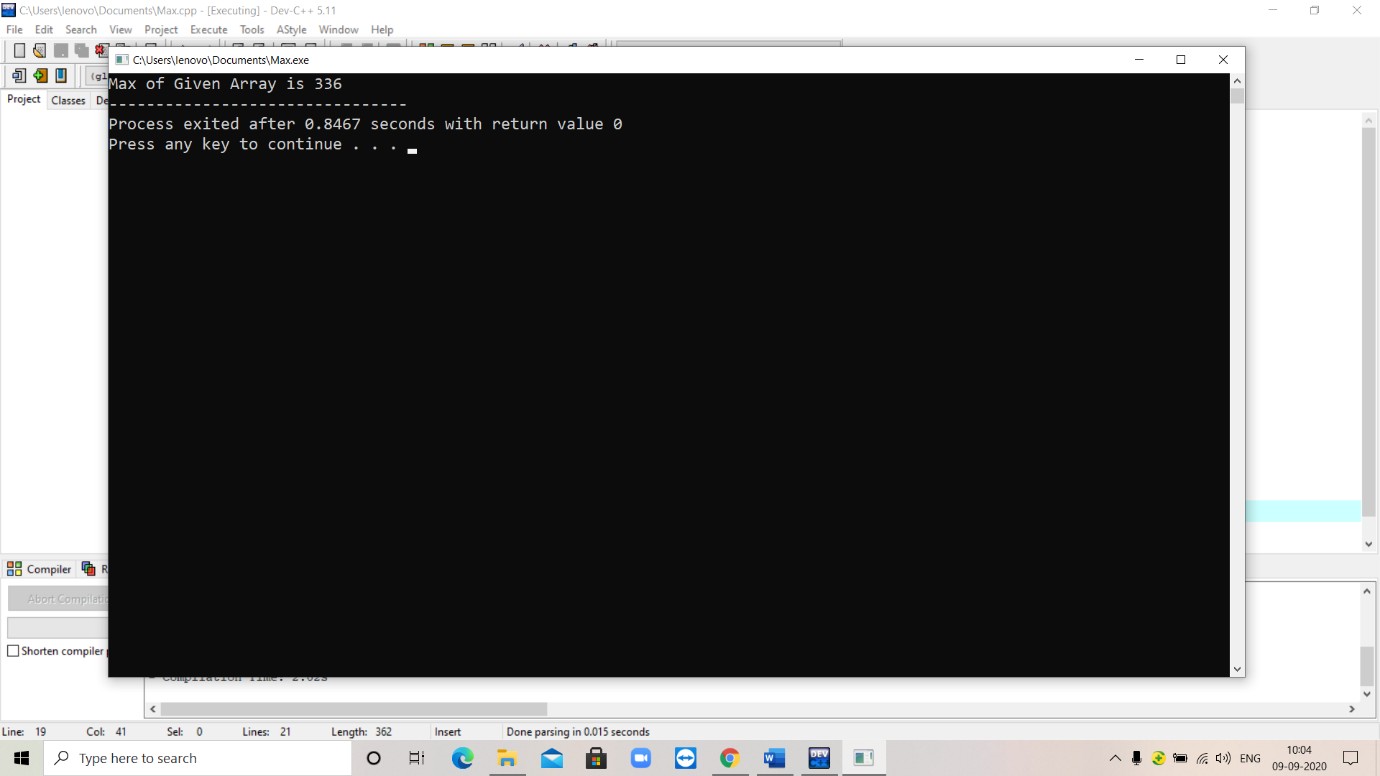
int n=sizeof(a)/sizeof(a[0]);

cout<<"Max of Given Array is"<<" "<<findmax(a,n);

return 0;

}

**OUTPUT:**



**AIM:**

Write a program to find whether the given array is unique or not

**ALGORITHM:**

**1.**Create Function unique

**2.**Put For Loop and Give Condition whether given array is unique or not

**3.**Create main function

**4.**Enter Array Elements

**5.**Check Given Array is unique or not

**PROGRAM:**

#include<iostream> using namespace std; int unique(int a[],int n)

{

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

{

for(int j=0;j<=n-1;j++)

{

if(a[i]==a[j])

return 0;

}

}

return 1;

}

int main()

{

int a[5],n;

cout<<"Enter Number of Elements";

cin>>n;

cout<<"Enter the inputs"; for(int i=0;i<n;i++) cin>>a[i];

if(unique(a,n))

cout<<"The Array is unique";

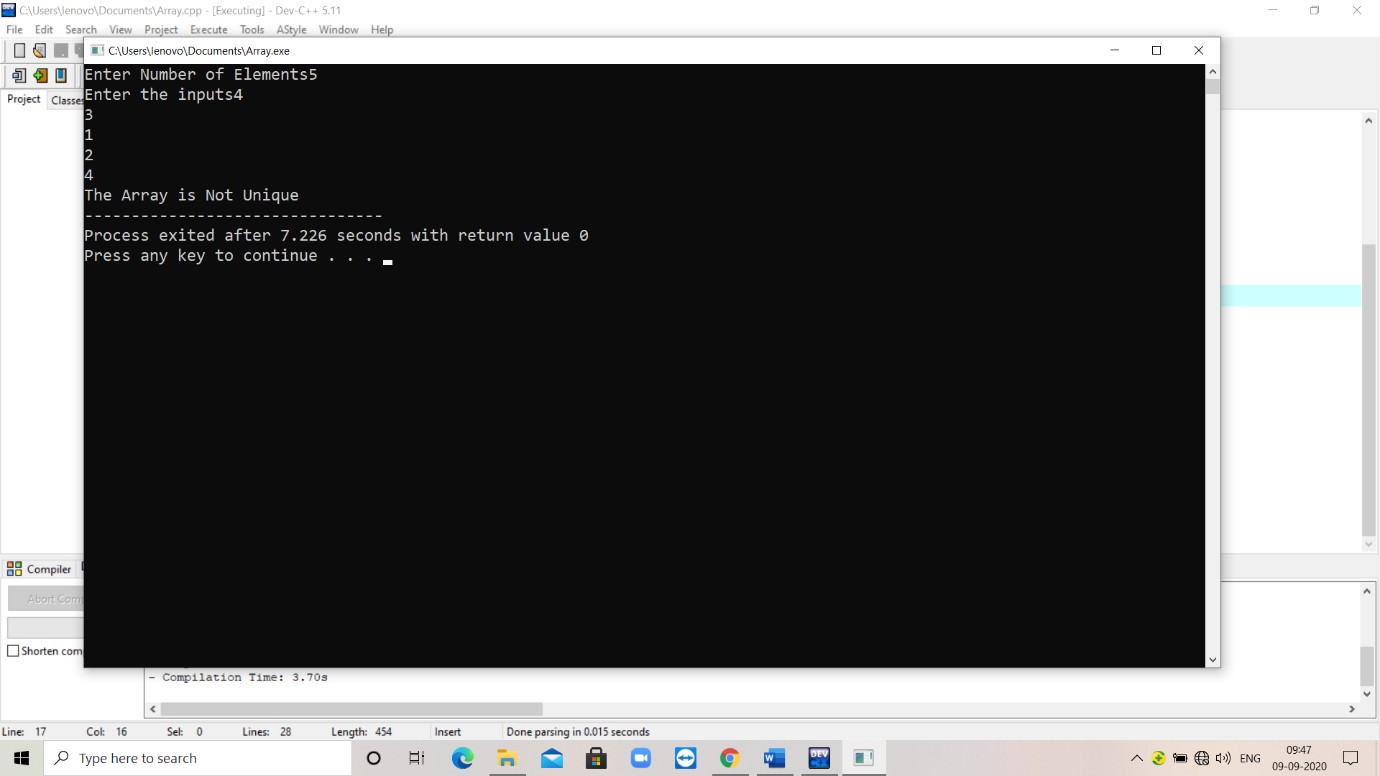
else

cout<<"The Array is Not Unique";

return 0;

}

**OUTPUT:**



**AIM:**

Write a program for matrix multiplication

**ALGORITHM:**

**1.**Enter first matrix element

**2.**Enter second matrix element

**3.**For Matrix Multiplication

Condition::mat3[i][j] +=(mat1[i][k] \* mat2[k][j])

**4.**Print the Matrix Muliplication of Given matrix

**PROGRAM:**

#include<iostream> using namespace std; int main()

{

int mat1[3][3],mat2[3][3],mat3[3][3]; cout<<"Enter Matrix 1 Elements\n"; for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

cin>>mat1[i][j];

}

}

cout<<"Enter Matrix 2 Elements\n";

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

cin>>mat2[i][j];

}

}

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

for(int k=0;k<3;k++)

{

mat3[i][j]=0;

mat3[i][j] +=(mat1[i][k] \* mat2[k][j]);

}

}

}

cout<<"Matrix Multiplication of Given Matrix is";

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

cout<<mat3[i][j]<<" ";

}

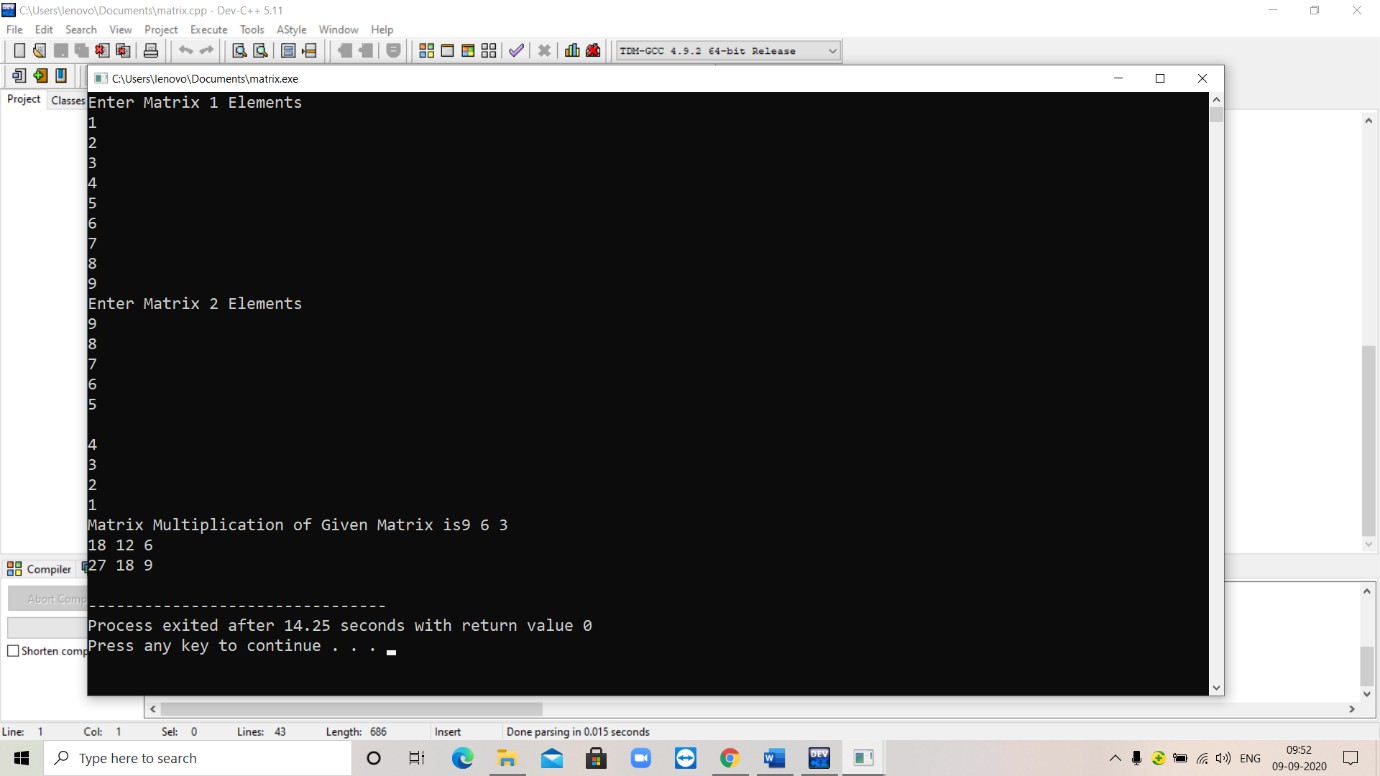
cout<<"\n";

}

return 0;

}

**OUTPUT:**



**AIM:**

Write a program to find number of digits in binary in non-recursive method

**ALGORITHM:**

**1.**Create a function binarydec with int a

**2.**Put “while” loop

**3.**And Give Condition a=a/2,count++.

**4.**Create main function

**5.**Enter a value

**6.**Print The no of digits in binary is binarydec(a)

**PROGRAM:**

#include<iostream> using namespace std; int binarydec(int a)

{

int count=1;

while(a>1)

{

a=a/2;

count++;

}

return count;

}

int main()

{

int a;

cout<<"Enter a value";

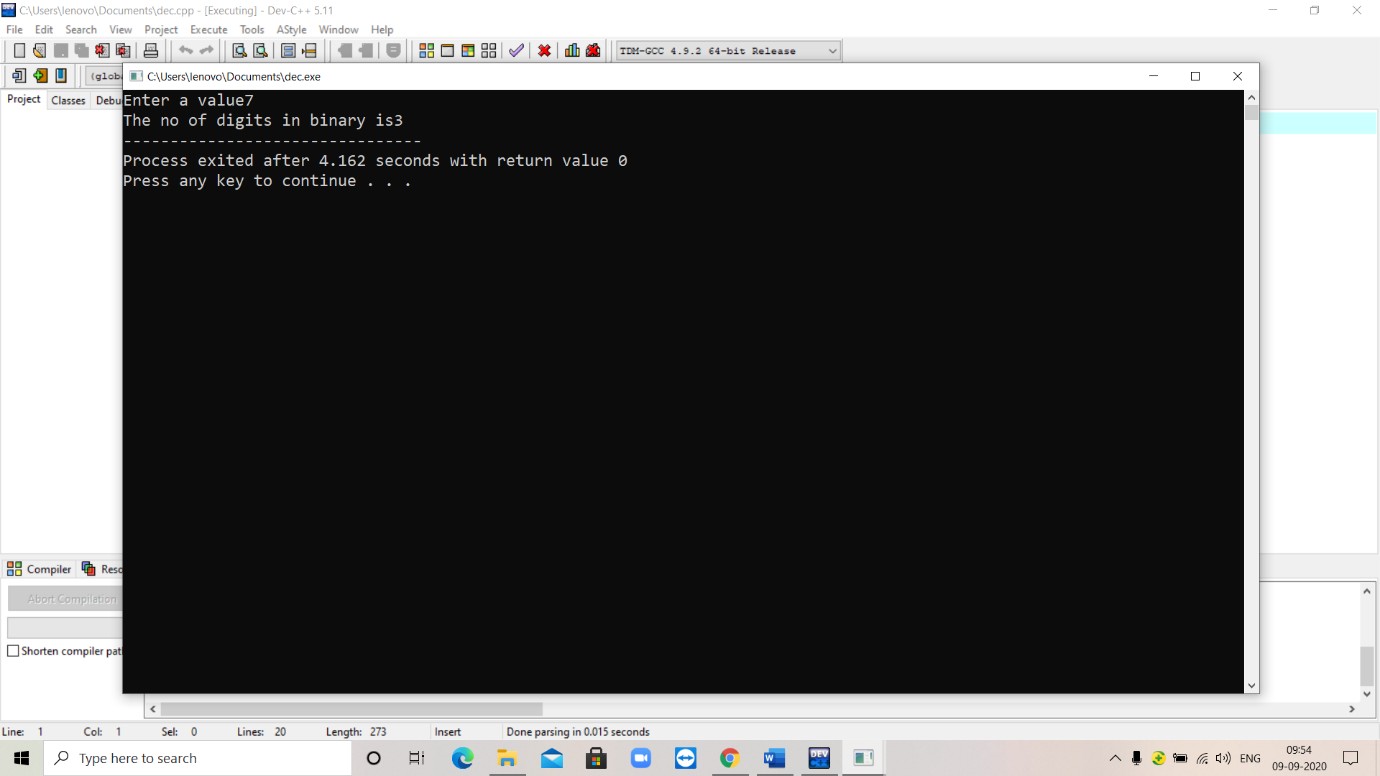
cin>>a;

cout<<"The no of digits in binary is"<<binarydec(a);

return 0;

}

**OUTPUT:**



**EX NO:3**

**AIM:**

Write the Program to find the factorial of Given Number

**ALGORITHM:**

**1.**Create a function fact with int n

**2.**Put “if” loop

**3.**if(n==1) return 1 else return n\*fact(n-1)

**4.**Create main function

**5.**Enter n value

**6.**Print Factorial of Given Number is fact(n)

**PROGRAM:**

#include<iostream> using namespace std; int fact(int n)

{

if(n==1) return 1; else

return n \* fact(n-1);

}

int main()

{

int n;

cout<<"Enter n Value\n";

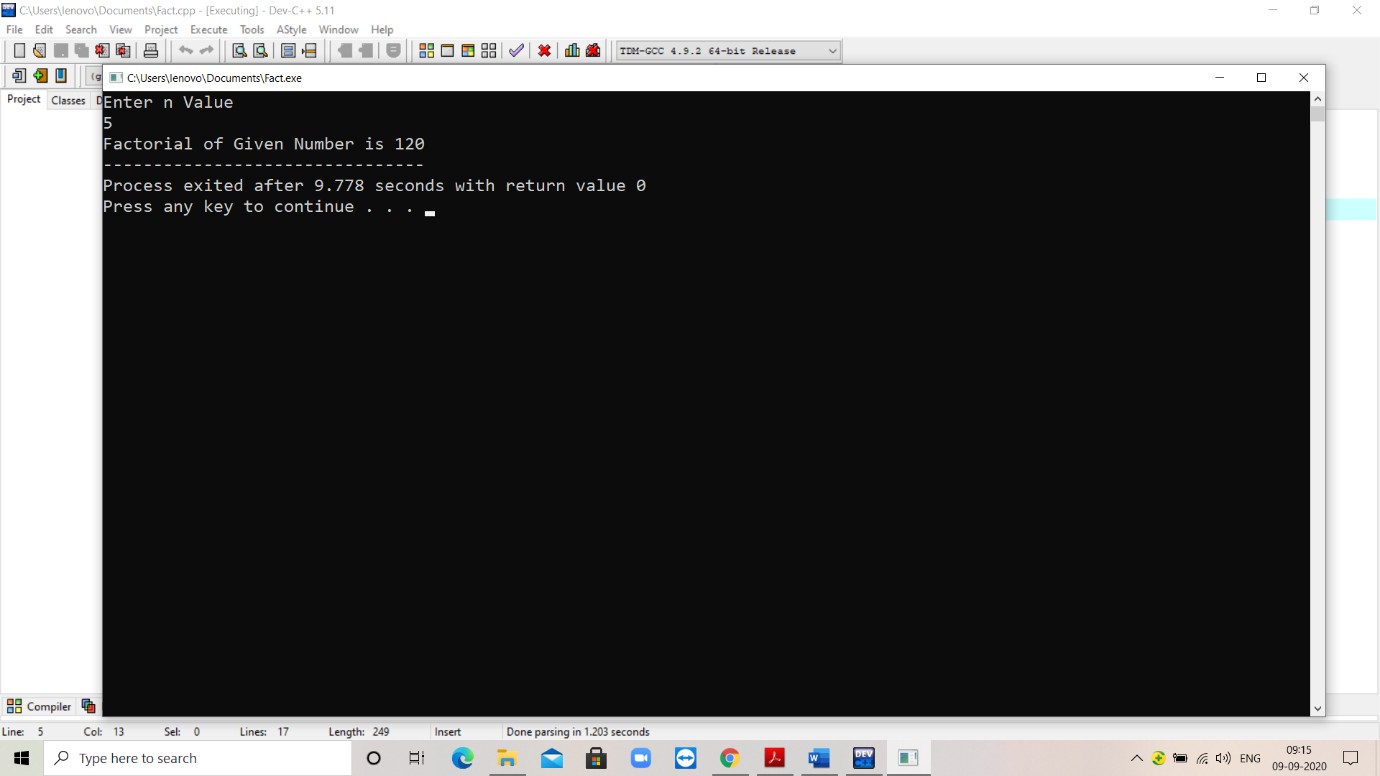
cin>>n;

cout<<"Factorial of Given Number is"<<" "<<fact(n);

return 0;

}

**OUTPUT:**



**AIM:**

Write the program to find Fibonacci Series

**ALGORITHM:**

**1.**Create a function fib with int n

**2.**Put “if” loop

**3.**Check initial condition if(n==0) return 0 else if(n==1) return 1 else return fib(n-1)+fib(n-2)

**4.**Create main function

**5.**Enter n value

**6.**Print Fibonacci Series of Given Number is fib(n)

**PROGRAM:**

#include<iostream> using namespace std; int fib(int n)

{

if(n==0)

return 0;

else if(n==1)

return 1;

else

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

}

int main()

{

int n;

cout<<"Enter n Value";

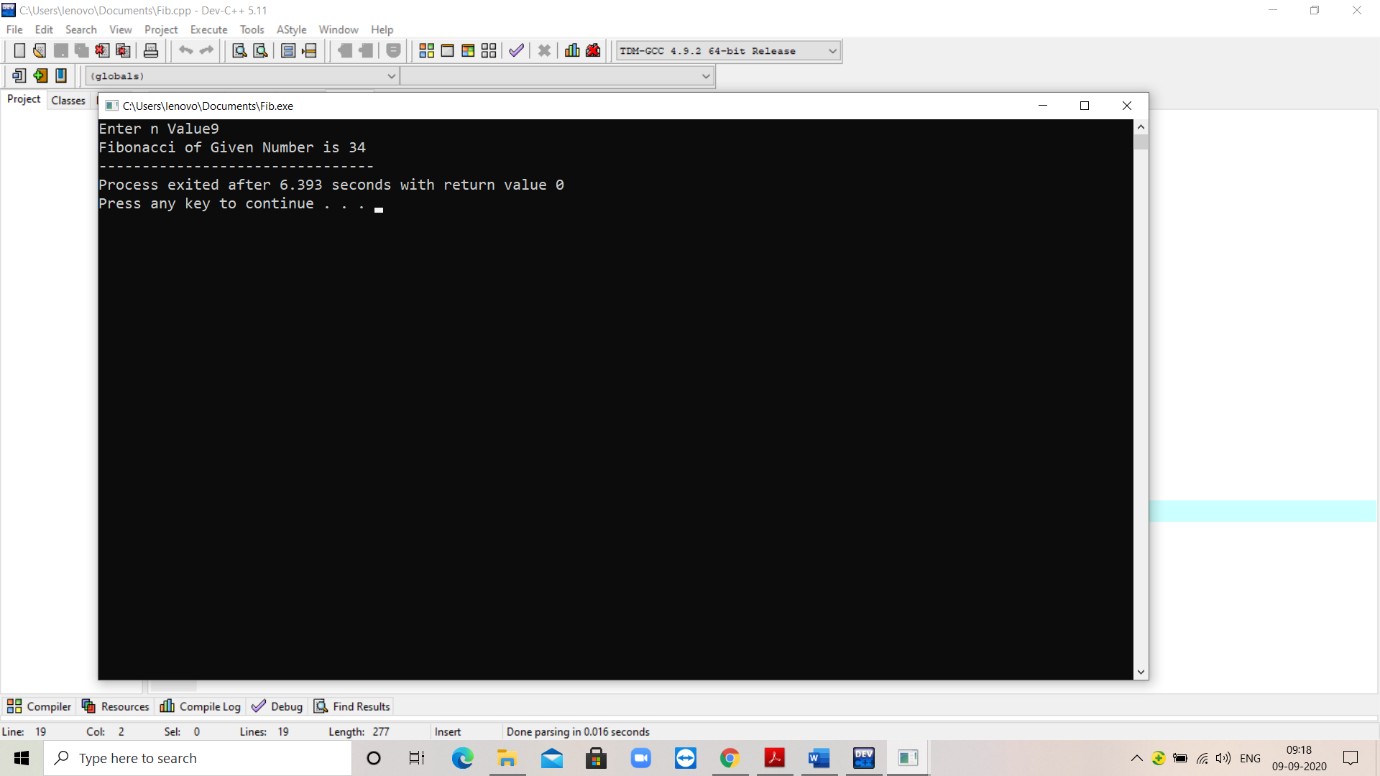
cin>>n;

cout<<"Fibonacci of Given Number is"<<" "<<fib(n);

return 0;

}

**OUTPUT:**



**AIM:**

Write a program to find number of digits in binary

**ALGORITHM:**

**1.**Create a function recbinarydec with int f

**2.**Put “if” loop

**3.**if(f==1) return 1 else return recbinarydec(f/2)+1;

**4.**Create main function

**5.**Enter f value

**6.**Print The no of digits in binary is recbinarydec(f)

**PROGRAM:**

#include<iostream> using namespace std; int recbinarydec(int f)

{

if(f==1) return 1; else

return recbinarydec(f/2)+1;

}

int main()

{

int f;

cout<<"Enter f value";

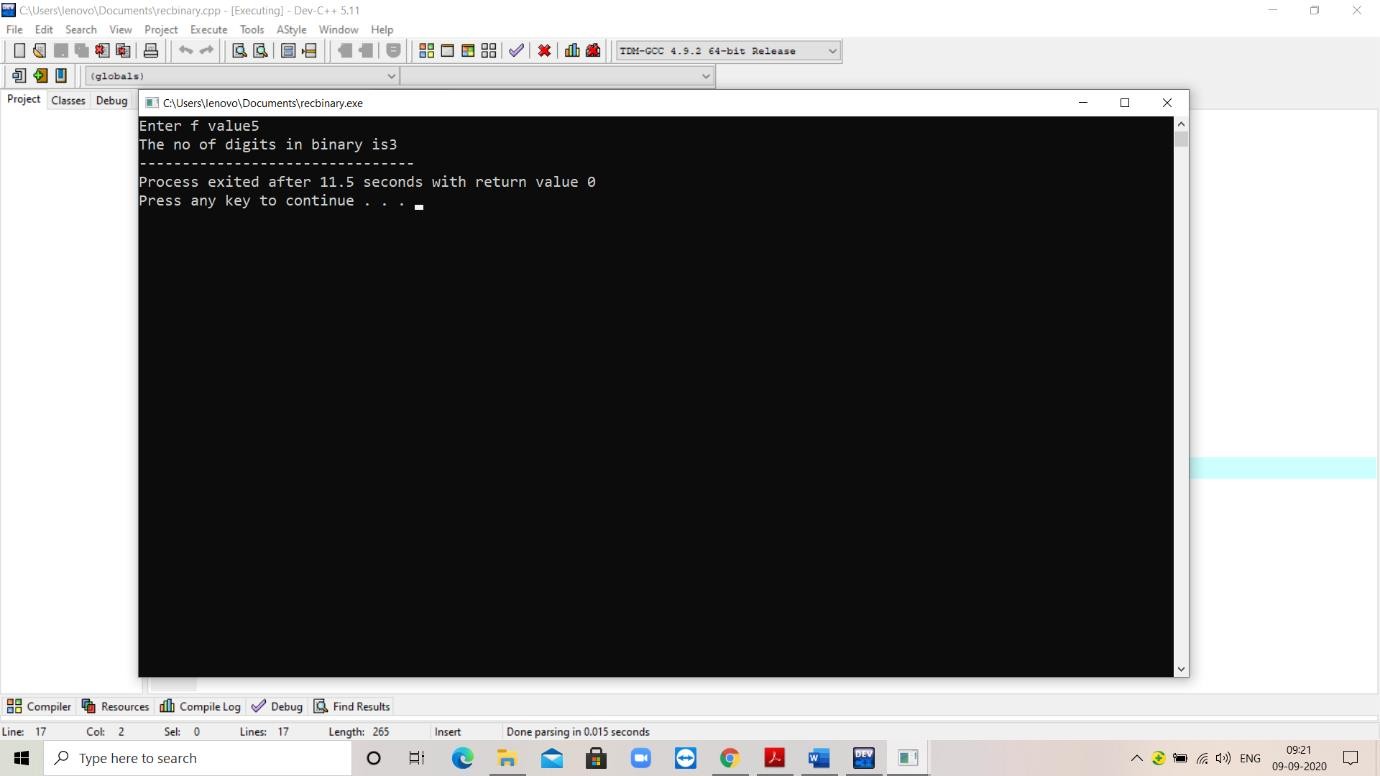
cin>>f;

cout<<"The no of digits in binary is"<<recbinarydec(f);

return 0;

}

**OUTPUT:**



**AIM:**

To implement Tower of Hanoi

**ALGORITHM:**

**1.**Create a function TOH with int n

**2.**Put “if” Loop

**3.**if(n==1) move the disc from “from tower” to the “to tower”

**4.**else call function TOH for “n-1”.Recall the function TOH “n-1”

disc and move it from “from tower” to “to tower”

**5.**Recall function again until the number of disc is 1.

**6.**Create Main Function

**7.**Enter No Of Rings

**PROGRAM:**

#include<iostream>

using namespace std;

void TOH(int n,char source,char dest,char inter)

{

if(n==1)

{

cout<<"Move ring 1 from tower"<<source<<"to tower"<<dest<<"\n";

return;

}

TOH(n-1,source,inter,dest);

cout<<"Move ring"<<n<<"from tower"<<source<<"to tower"<<dest<<"\n";

TOH(n-1,inter,dest,source);

}

int main()

{

int n;

cout<<"Enter Number of Rings";

cin>>n; TOH(n,'A','C','B'); return 0;

}

**OUTPUT:**

