**ADA LAB-1**

* **ITERATIVE GCD**
  + - **WRITTEN PROGRAM**

#include <iostream>

using namespace std;

int hcf(int a, int b)

{

while (a != b)

{

if (a > b)

a = a - b;

else

b = b - a;

}

return a;

}

int main()

{

int a,b;

cout<<"enter first number: ";

cin>>a;

cout<<"enter second number: ";

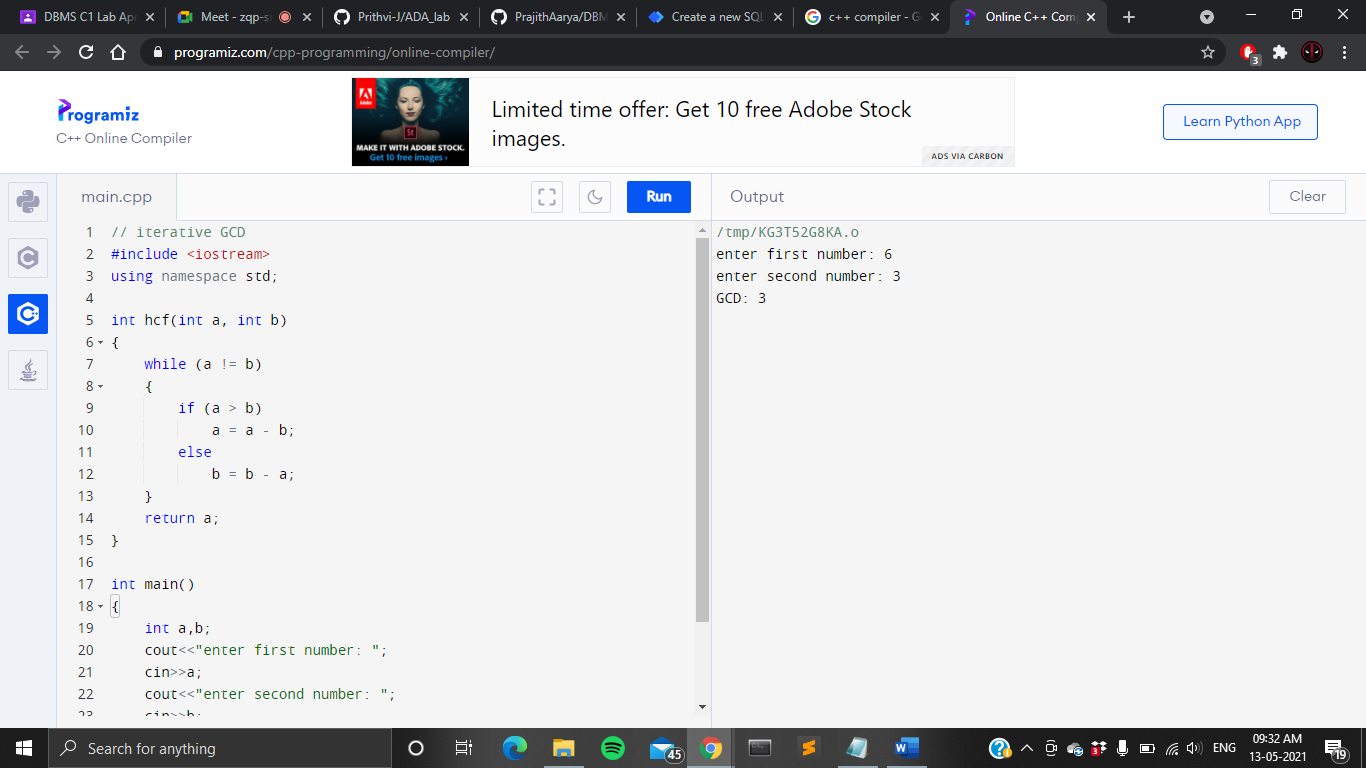
cin>>b;

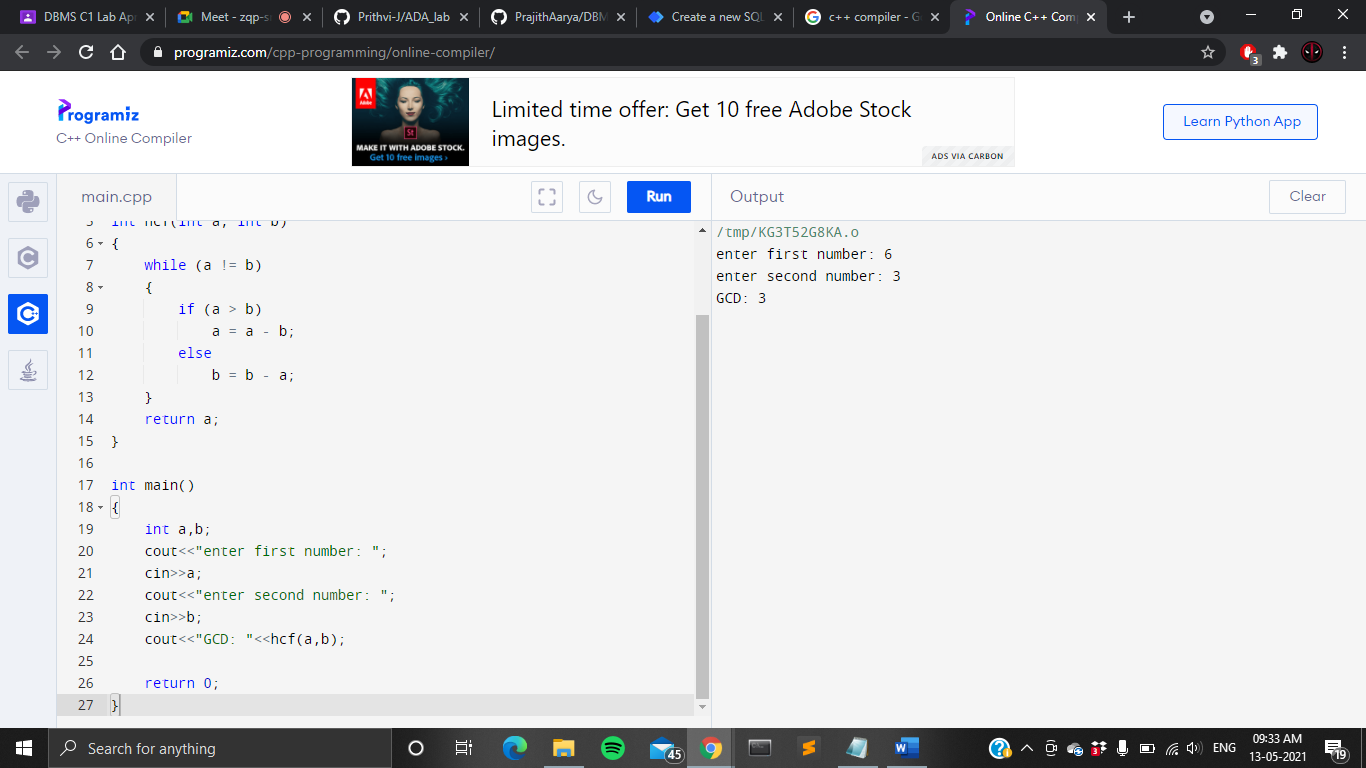
cout<<"GCD: "<<hcf(a,b);

return 0;

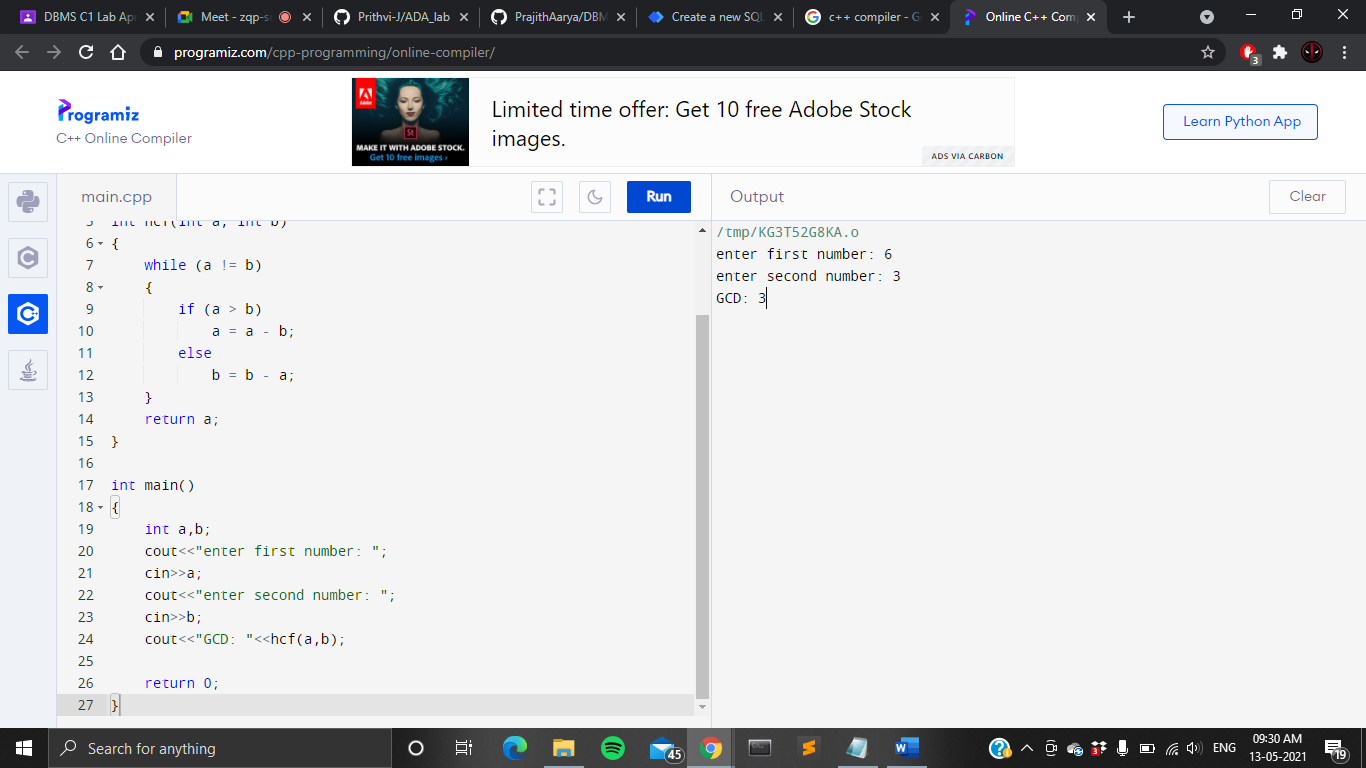
}

* + - **PROGRAM**





* + - **OUTPUT**



* **RECURSIVE GCD**
  + - **WRITTEN PROGRAM**

//recursive GCD

#include<iostream>

using namespace std;

int gcd(int a, int b)

{

if (a == 0 || b == 0)

return 0;

else if (a == b)

return a;

else if (a > b)

return gcd(a-b, b);

else

return gcd(a, b-a);

}

int main()

{

int a,b;

cout<<"enter first number: ";

cin>>a;

cout<<"enter second number: ";

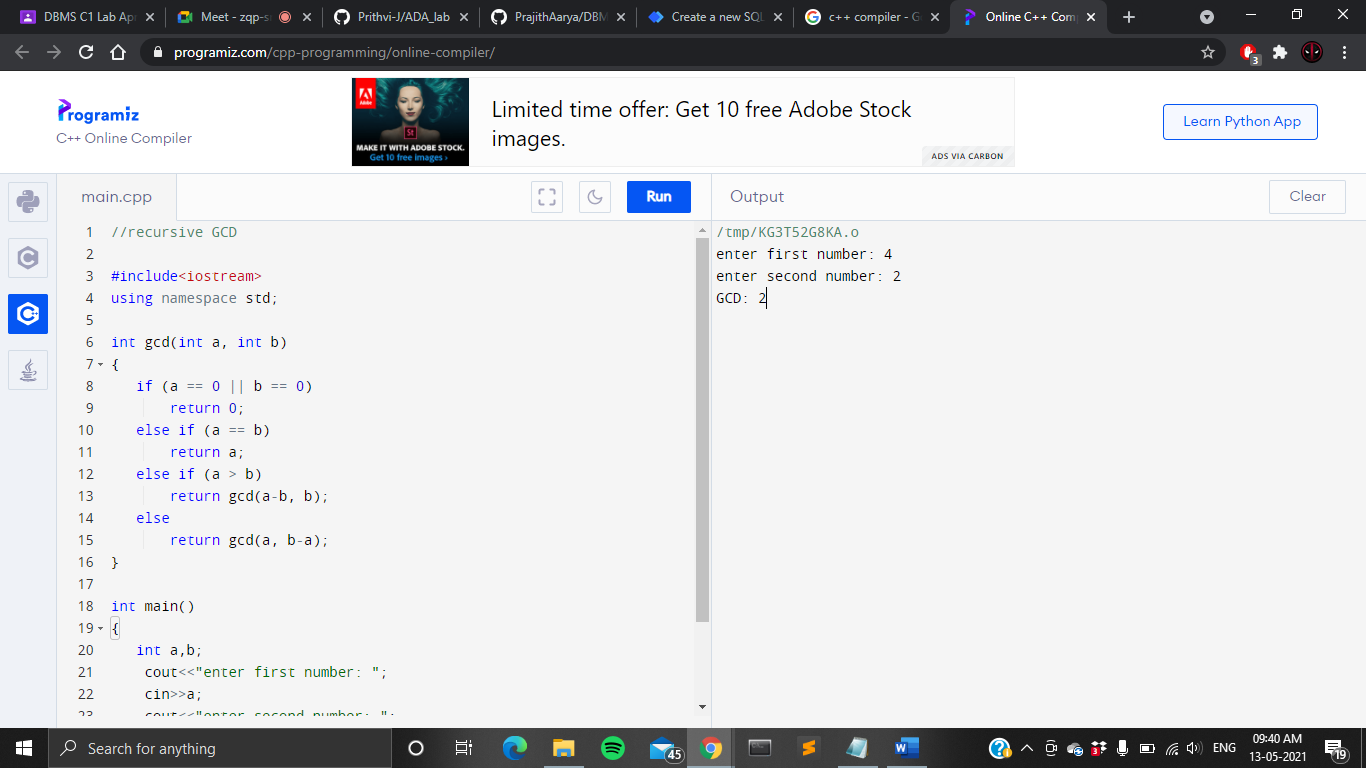
cin>>b;

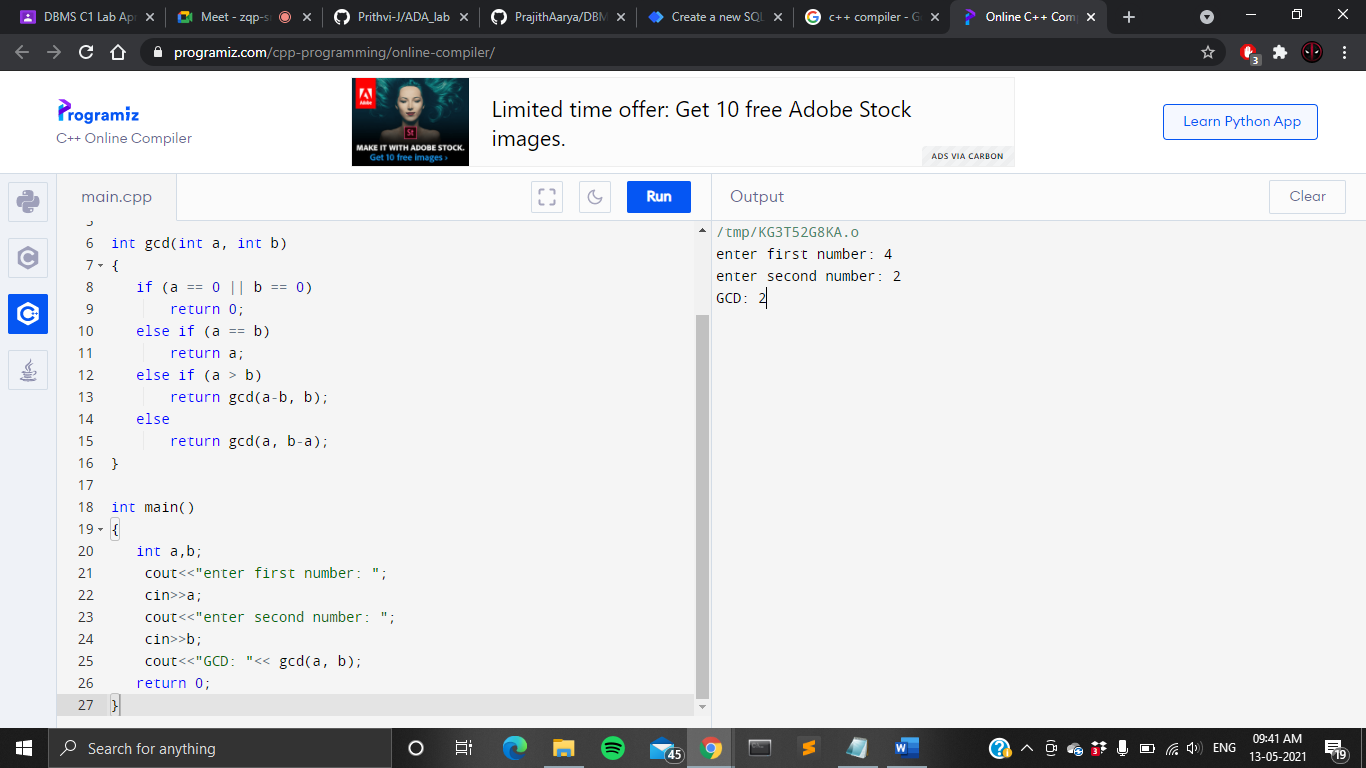
cout<<"GCD: "<< gcd(a, b);

return 0;

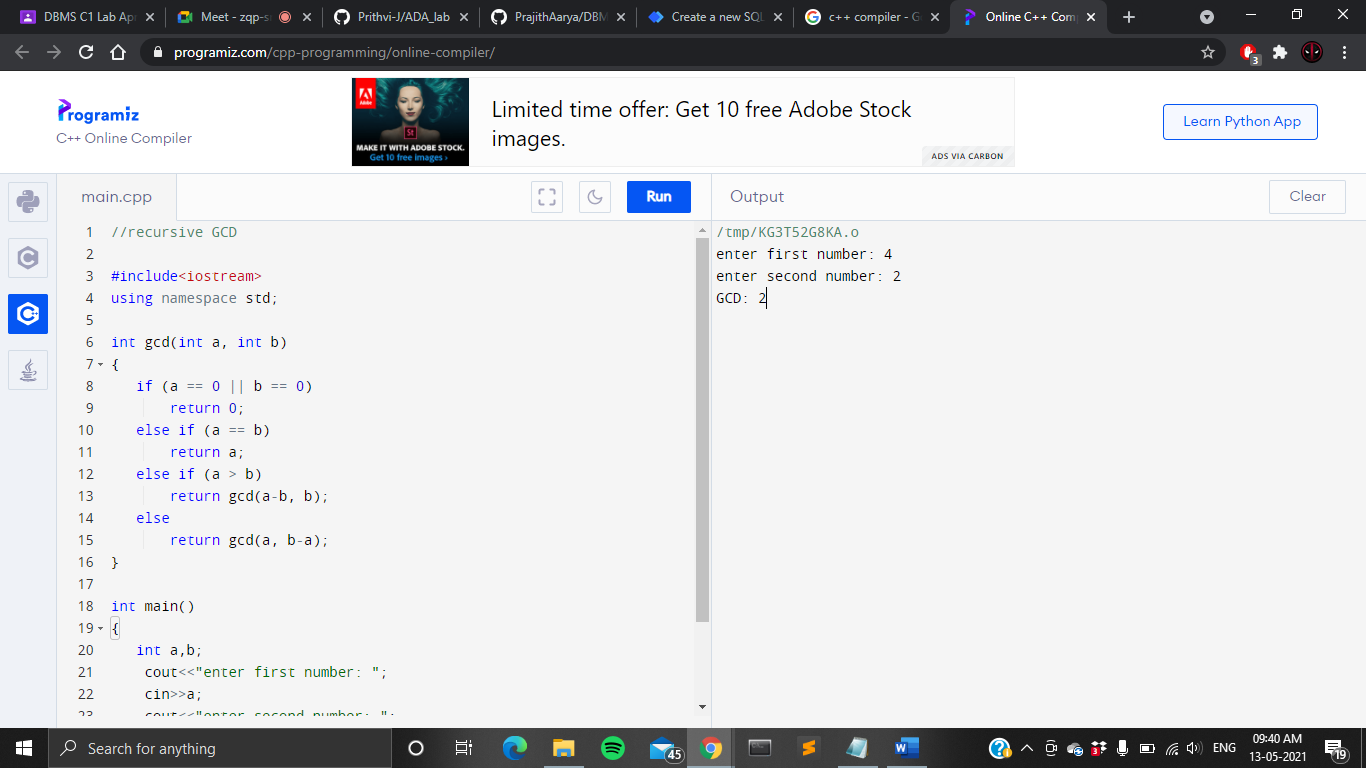
}

* + - **PROGRAM**





* + - **OUTPUT**



* + - * **RECURSIVE TOWER OF HANOI**

**WRITTEN PROGRAM**

//recursive tower of hanoi

#include <iostream>

using namespace std;

void towerOfHanoi(int n, char from\_rod,char to\_rod, char aux\_rod)

{

if (n == 1)

{

cout << "Move disk 1 from rod " << from\_rod <<" to rod "<<to\_rod<<endl;

return;

}

towerOfHanoi(n - 1, from\_rod, aux\_rod, to\_rod);

cout << "Move disk " << n << " from rod " << from\_rod <<" to rod " << to\_rod <<endl;

towerOfHanoi(n - 1, aux\_rod, to\_rod, from\_rod);

}

int main()

{

int n;

cout<<"enter the number of discs: ";

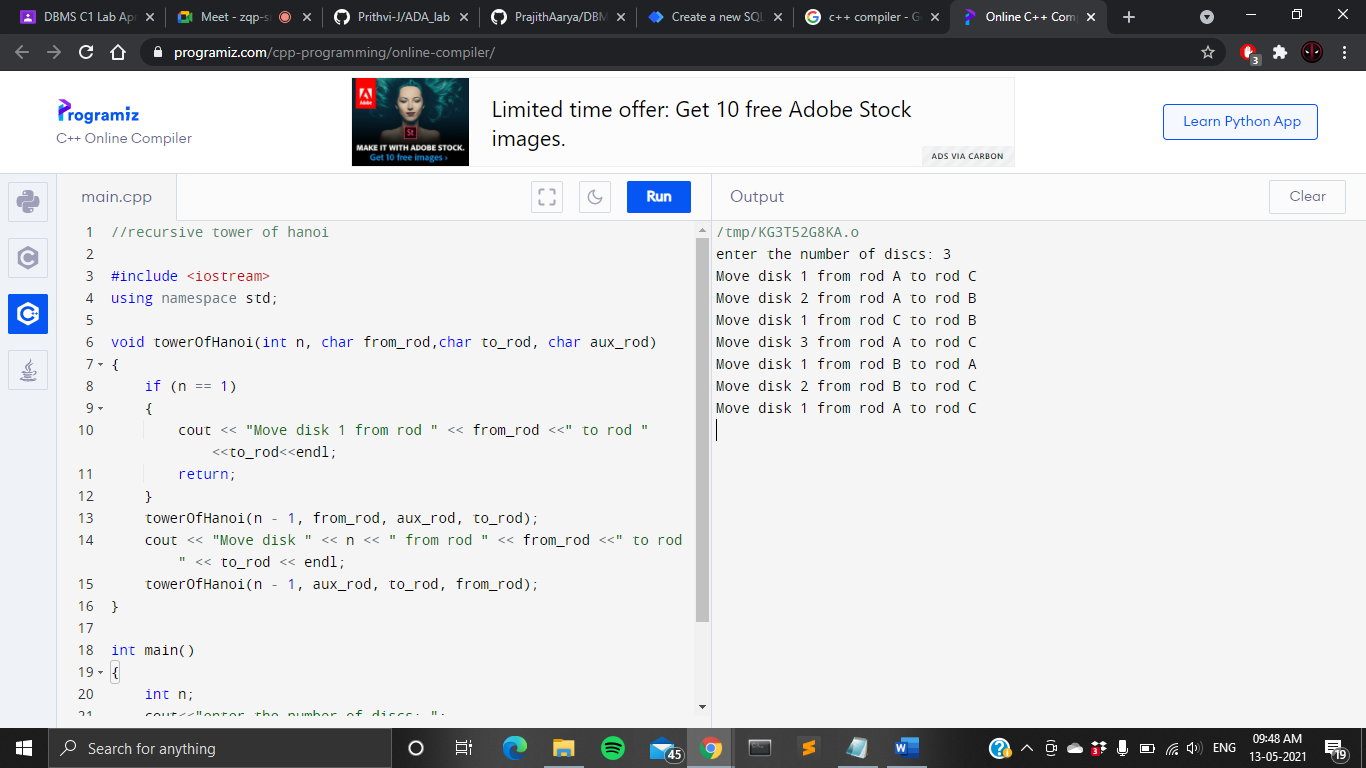
cin>>n;

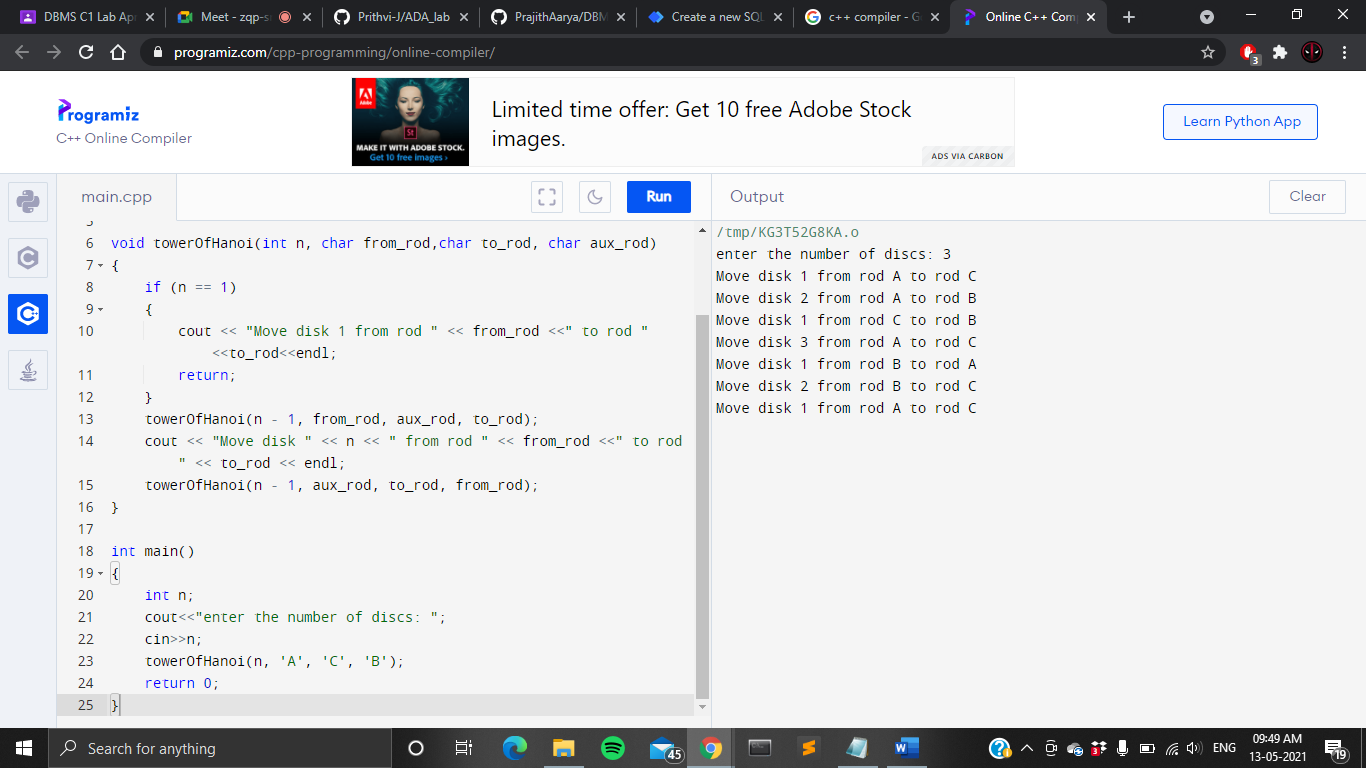
towerOfHanoi(n, 'A', 'C', 'B');

return 0;

}

**PROGRAM**





**OUTPUT**

