**LAB 1 PRACTICE TASKS:  
FIND GREATEST COMMON DIVISOR:**

**CODE:**

#include<iostream>

using namespace std;

void main()

{

int n1, n2;

cout << " enter two numbers : " << endl;

cin >> n1 >> n2;

int gcd = 1;

int k = 2;

while (k <= n1 && k <= n2)

{

if (n1 % k == 0 && n2 % k == 0)

{

gcd = k;

}

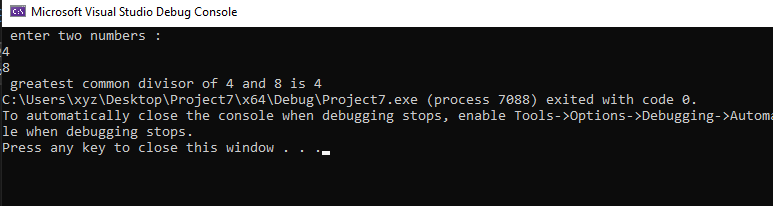
k++;

}

cout << " greatest common divisor of " << n1 << " and " << n2 << " is " << gcd;

}

**OUTPUT:**



**LAB 1:**

**PRACTICE TASK 1:**

**CODE:**

#include<iostream>

using namespace std;

void main()

{

int hrs\_work;

int hrs\_rate;

int weekly\_gross = 0;

int extra\_hr;

cout << " enter your worked hours " << endl;

cin >> hrs\_work;

cout << " enter hours rate " << endl;

cin >> hrs\_rate;

if (hrs\_work == 40)

{

weekly\_gross = hrs\_work \* hrs\_rate;

cout << " total gross salary of a week while working for " << hrs\_work << " hours " << weekly\_gross;

}

else if (hrs\_work < 40)

{

weekly\_gross = hrs\_work \* hrs\_rate;

cout << " total gross salary of a week while working for " << hrs\_work << " hours " << weekly\_gross;

}

else if (hrs\_work > 40)

{

//int extra\_hr;

extra\_hr = hrs\_work - 40;

int r = 1.3 \* hrs\_rate;

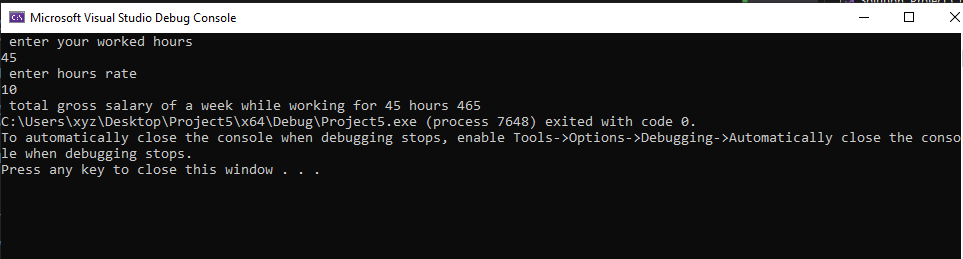
int overtime = (r \* extra\_hr) + (hrs\_rate\*40);

cout << " total gross salary of a week while working for " << hrs\_work << " hours " << overtime;

}

}

**OUTPUT:**

****

**PRACTICE TASK 2:**

**CODE:**

#include<iostream>

using namespace std;

void main()

{

int limit;

cout << " set the limit for loop excution : ";

cin >> limit;

cout << " your loop will excecute from 1 to " << limit << endl;

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

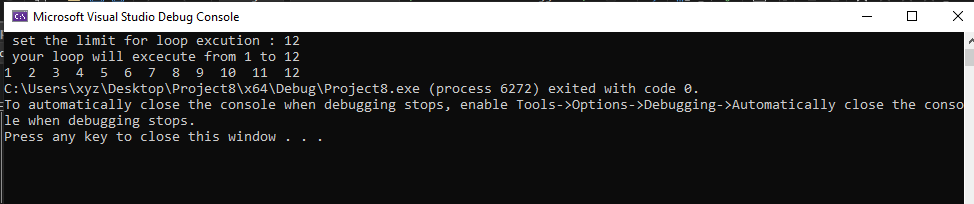
{

cout << i << " ";

}

}

**OUTPUT:**

****

**PRACTICE TASK 3:**

**CODE:**

#include<iostream>

using namespace std;

void main()

{

int n;

cout << " enter a number to find whether it is prime of not : " << endl;

cin >> n;

int flag = 1;

for (int k = 2;k <= n / 2;k++)

{

if (n % k == 0)

{

flag = 0;

break;

}

}

if (flag == 1)

{

cout << " prime number " << endl;

}

else

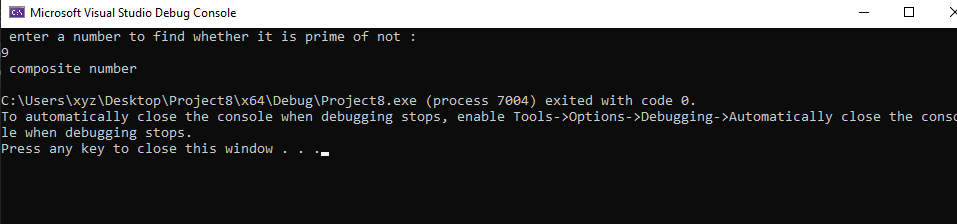
{

cout << " composite number " << endl;

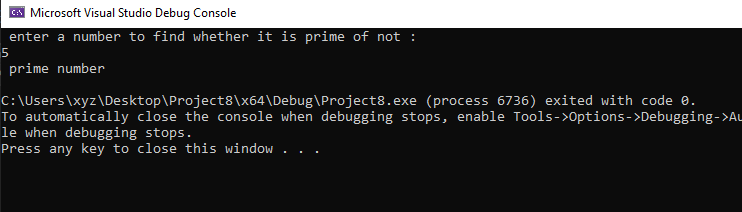
}

}

**OUTPUT:**

****

**2:**

****

**Task 1: lab 3:**

**Code:**

**Output:**

#include<iostream>

using namespace std;

class Android\_Device

{

int IMEIno;

string type;

string make;

int Modelno;

float memory;

string O\_S;

//constructor

public:

Android\_Device()

{

IMEIno = 0;

type = " ";

make = " ";

Modelno = 0;

memory = 0.0;

O\_S = " ";

}

void set\_values(int a, string b, string c, int d, float e, string f)

{

IMEIno = a;

type = b;

make = c;

Modelno = d;

memory = e;

O\_S = f;

}

void display()

{

cout << " IMEI number : " << IMEIno << endl;

cout << " type : " << type << endl;

cout << " make : " << make << endl;

cout << " model number : " << Modelno << endl;

cout << " memory : " << memory << endl;

cout << " operating system : " << O\_S << endl;

}

};

int main()

{

Android\_Device device1;

device1.set\_values(12345, "smartphone", "samsung", 123, 64.0, "os11");

device1.display();

return 0;

}