Experiment: 1

Student Name: Jitesh Kumar **UID: 20BCS2334**
Branch: CSE **Section/Group: 20BCS-WM-903/A**
Semester: 5th
Subject Name: DAA Lab **Subject Code:** 21-CSP-312

1. Aim/Overview of the practical:

Code and analyze to compute the greatest common divisor (GCD) of two numbers.

Example: GCD of 20 and 30 is 10 (As, 10 is the largest number which divides 20 & 30 both with remainder 0).

2. Task to be done/ Which logistics used:

To find GCD of two numbers.

3. Algorithm/Flowchart (For programming based labs):

Pseudo Code of the Algorithm-
Step 1: Let a, b be the two numbers.
Step 2: a mod b = R.
Step 3: Let a = b and b = R.
Step 4: Repeat Steps 2 and 3 until a mod b is greater than 0.
Step 5: GCD = b.
Step 6: Finish.

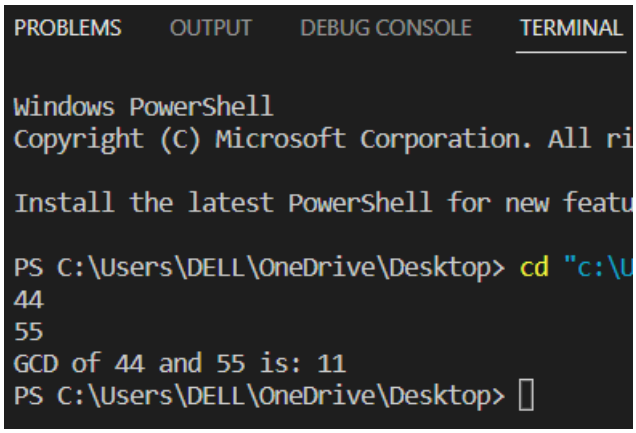
4. Steps for experiment/practical/Code:

```
#include<bits/stdc++.h>
using namespace std;
int gcd(int x,int y)
{
if(y==0)
   return x;
else
   return gcd(y,x%y);
}
int main()
{
   int a,b;
   cin>>a>>b;
   cout<<"GCD of "<<a<<" and "<<b<<" is: "<<gcd(a,b);
   return 0;
}
```

5. Observations/Discussions/ Complexity Analysis:

Time complexity of finding GCD of two number using Euclidean method is O(log n).

6. Result/Output/Writing Summary:



Learning outcomes (What I have learnt):

1. To know how Euclidean algorithm works.
2. To learn how to use recursion for solving problems.