

Name: Rutikesh Sawant

Batch: B2

Subject: CNS Lab

PRN: 2019BTECS00034

Assignment 10

Aim: Find the GCD of two given numbers using Extended Euclidean Algorithm

Theory:

In arithmetic and computer programming, the extended Euclidean algorithm is an extension to the Euclidean algorithm, and computes, in addition to the greatest common divisor (gcd) of integers a and b , also the coefficients of Bézout's identity, which are integers x and y such that. The extended Euclidean algorithm also refers to a very similar algorithm for computing the polynomial greatest common divisor and the coefficients of Bézout's identity of two univariate polynomials.

Code:

```
#include <bits/stdc++.h>
using namespace std;

void file()
{
#ifdef ONLINE_JUDGE
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
}

int ansS, ansT;

int findGcdExtended(int r1, int r2, int s1, int s2, int t1, int t2)
{
    // Base Case
    if (r2 == 0)
```

```

{
    ansS = s1;
    ansT = t1;
    return r1;
}

int q = r1 / r2;
int r = r1 % r2;

int s = s1 - q * s2;
int t = t1 - q * t2;

    cout << q << " " << r1 << " " << r2 << " " << r << " " << s1 << " " << s2 << " " << s << " " << t1
<< " " << t2 << " " << t << endl;

    return findGcdExtended(r2, r, s2, s, t2, t);
}

int main()
{
    file();

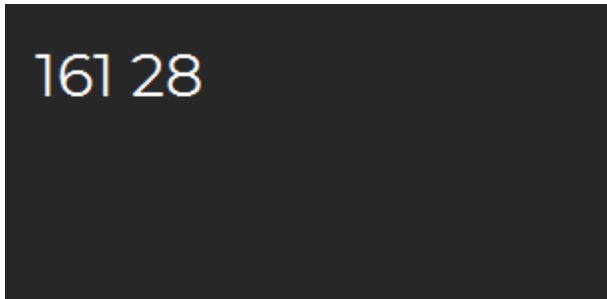
    int num1, num2, s, t;
    cout << "Enter 2 numbers to find GCD" << endl;
    cin >> num1 >> num2;

    int gcd = findGcdExtended(num1, num2, 1, 0, 0, 1);
    cout << "\n\nGCD = " << gcd << endl;
    cout << "S = " << ansS << endl;
    cout << "T = " << ansT << endl;

    return 0;
}

```

Output:



161 28

Enter 2 numbers to find GCD

5 161 28 21 1 0 1 0 1 -5

1 28 21 7 0 1 -1 1 -5 6

3 21 7 0 1 -1 4 -5 6 -23

GCD = 7

S = -1

T = 6