19SE02IT058

Extended Euclidean Algorithm

Code:

```
#include <iostream>
#include <tuple>
using namespace std;



tuple<int, int, int> extended_gcd(int m, int n)
{
   if (m== 0) {
      return make_tuple(n, 0, 1);
   }

   int gcd, x, y;

   // unpack tuple returned by function into variables
   tie(gcd, x, y) = extended_gcd(n % m, m);

   return make_tuple(gcd, (y - (n/m) * x), x);
}

int main()
{
   int m = 35;
   int n = 100;

   tuple<int, int, int> t = extended_gcd(m, n);
```

```
    t
    int gcd = get<0>(t);
    int x = get<1>(t);
    int y = get<2>(t);



    cout << "The GCD is " << gcd << endl;
    cout << "x = " << x << " y = " << y << endl;
    cout << m << "*" << x << " + " << n << "*" << y << " = " << gcd << endl;


    return 0;
}
```

Output: