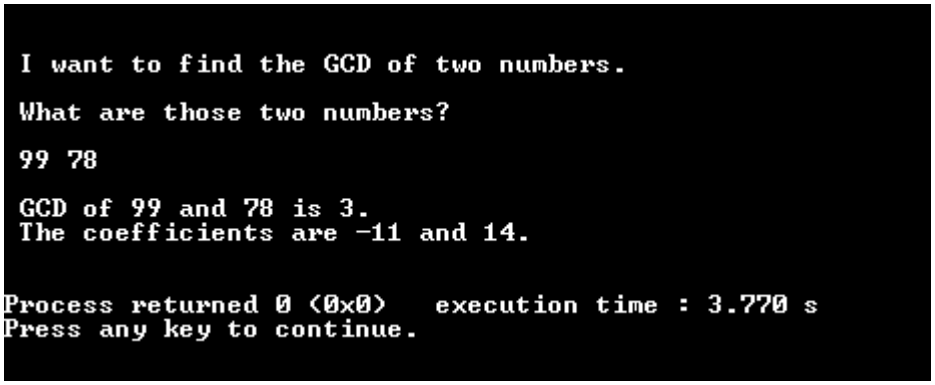
```
/*C Program to implement Extended-Euclidean Algorithm
Input : Two integers (a,b)
Output : GCD of a and b, coefficients x and y
          such that GCD = ax + by
*/

#include <stdio.h>
int extendedEuclid(int , int , int *, int *);
int main()
{
   int a, b, d, x, y, temp;

   printf("\n\n I want to find the GCD of two numbers.\n\n");
   printf(" What are those two numbers?\n\n ");
   scanf("%d%d",&a,&b);

   if(a<0) //Treat negative integer just like positive integer
   a = abs(a);
   //Either way the answer will be same because we are finding greatest
factor and it will be positive only.
   if(b<0)
   b = abs(b);
   if(a < b)
   {
      temp =a; //If a is less than b, change their position
      a=b;
      b= temp;
   }
```

```
    d = extendedEuclid(a, b, &x, &y);

    printf("\n GCD of %d and %d is %d. \n The coefficients are %d and
%d.\n\n ",a,b,d,x,y);

    return 0;
}

int extendedEuclid(int a, int b, int *x, int *y)
{
    if (a == 0)
    {
        *x = 0;
        *y = 1;
        return b;
    }

    int x1, y1; // To store results of recursive call
    int gcd = extendedEuclid(b%a, a, &x1, &y1);

    *x = y1 - (b/a) * x1;
    *y = x1;

    return gcd;
}
```

Sample input and output

1. When both a and b are +ve integers



2. When b = 0