Name: **Kritika Bhattad**

Roll no: **C1-06**

Subject: **Cryptography Lab**

Practical 6:

Aim: **Implement Euler's Phi-Function to support cryptography algorithms.**

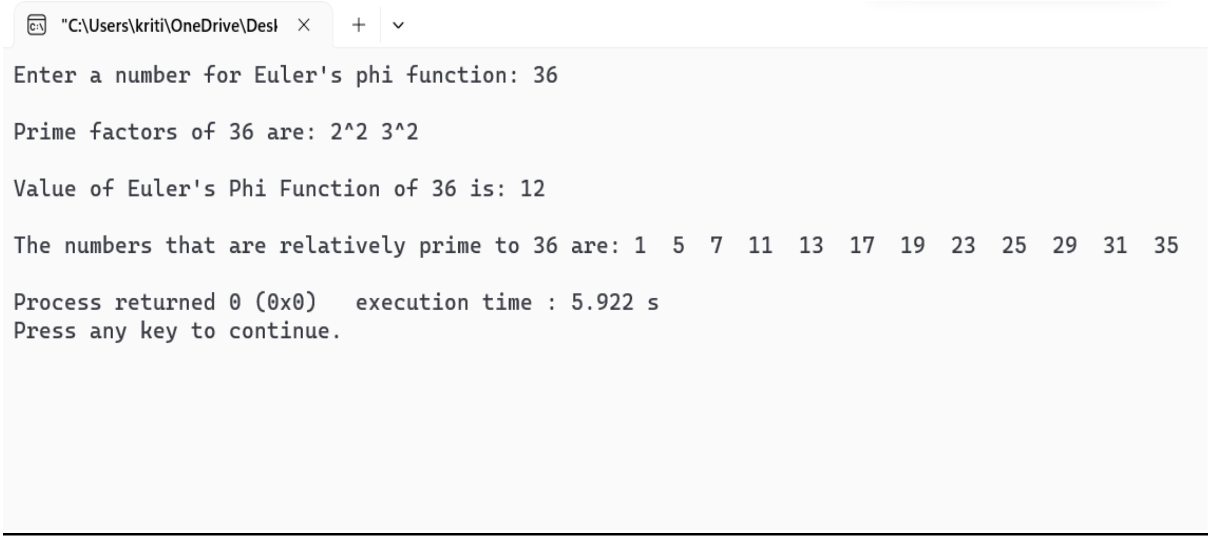
Code:

```
#include <stdio.h>

#include<math.h>

int p[100],e[100],k=0,l=0,power=0;


void factorization(int n){

   printf("\nPrime factors of %d are: ", n);

   int i;

   // Check for 2 as a prime factor

   while (n % 2 == 0) {

      power++;

      n /= 2;

   }

   if (power > 0){

      p[k++]=2;

      e[l++]=power;

      printf("2^%d ", power);

   }

   // Check for other odd prime factors

   for(i = 3; i * i <= n; i += 2) {

      power = 0;

      while (n % i == 0) {
          power++;
          n /= i;
      }
      if (power > 0){
          p[k++]=i;
          e[l++]=power;
          printf("%d^%d ", i, power);
      }
   }
   // If n is still greater than 2, it is a prime factor
   if (n > 2){
      p[k++]=n;
      e[l++]=1;
      printf("%d^1 ", n);
   }
   printf("\n");
}

int gcd(int a, int b){
   int i,gcd;
   for(i=1; i <= a && i <= b; ++i){
      if(a%i==0 && b%i==0)
          gcd = i;
   }
   if(gcd==1)
     return 1;
   else
     return 0;
}
```

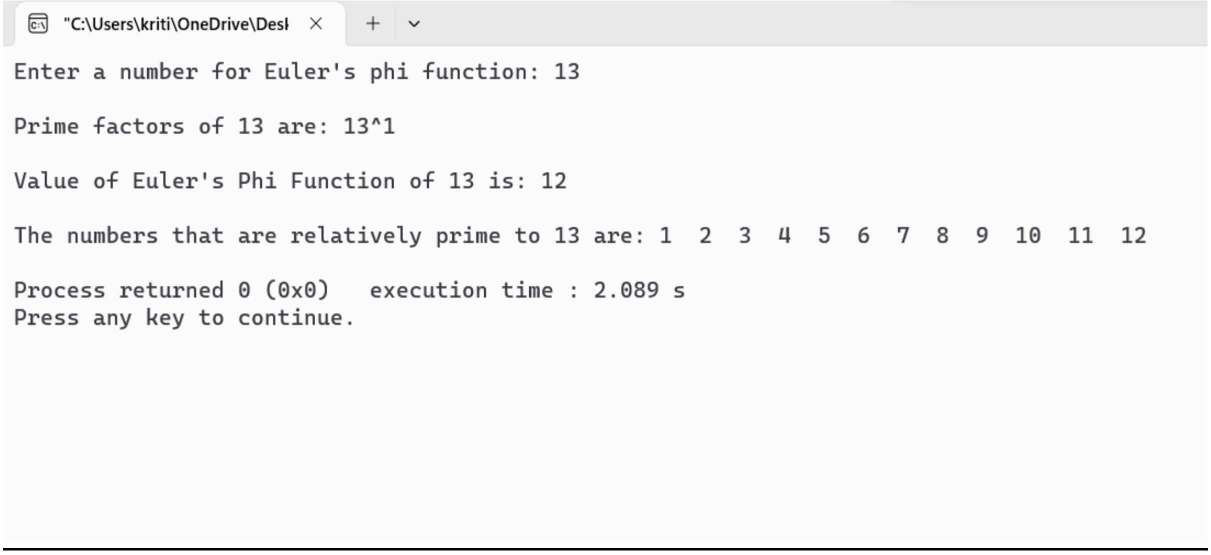
```
int prime_no(int n){
   int i,k;
   if(n == 0 || n == 1)
     k = 1;

   for(i = 2; i <= n / 2; ++i) {
    if(n % i == 0) {
      k = 1;
      break;
    }
   }

   if (k == 0)
    return 1;
   else
    return 0;
}



int main() {
   int i,n,phi_n=1;

   printf("Enter a number for Euler's phi function: ");
   scanf("%d", &n);

   //Prime factorization of n
   factorization(n);
```

```
    //Calculation of Euler's Phi Function for different types of input values of n
    if(n==1){
        phi_n=0;
    }
    else if(prime_no(n)){
        if(power>0)
            phi_n=pow(n,power) - pow(n,power-1);
        else
            phi_n=n-1;
    }
    else{
        for(i=0;i<k;i++){
            phi_n=phi_n * (pow(p[i],e[i]) - pow(p[i],e[i]-1));
        }
    }
    printf("\nValue of Euler's Phi Function of %d is: %d\n",n,phi_n);

    //All relatively prime numbers to n
    printf("\nThe numbers that are relatively prime to %d are: ",n);
    for(i=1;i<n;i++){
        if(gcd(i,n)==1){
            printf("%d  ",i);
        }
    }
    printf("\n");


    return 0;
}
```

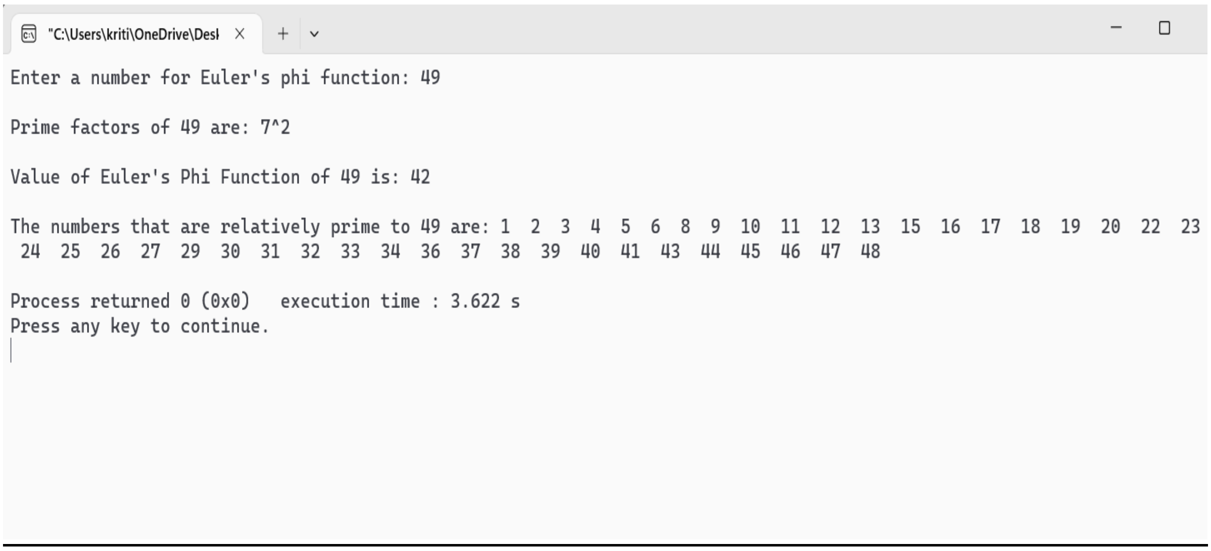
Output:



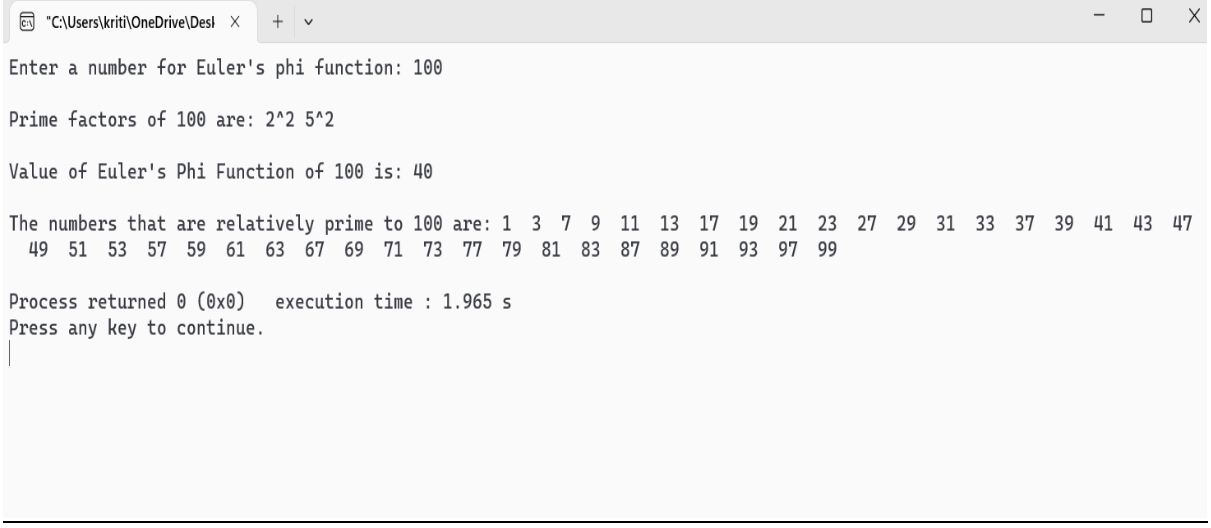
```
Enter a number for Euler's phi function: 36

Prime factors of 36 are: 2^2 3^2

Value of Euler's Phi Function of 36 is: 12

The numbers that are relatively prime to 36 are: 1  5  7  11  13  17  19  23  25  29  31  35

Process returned 0 (0x0)   execution time : 5.922 s
Press any key to continue.
```



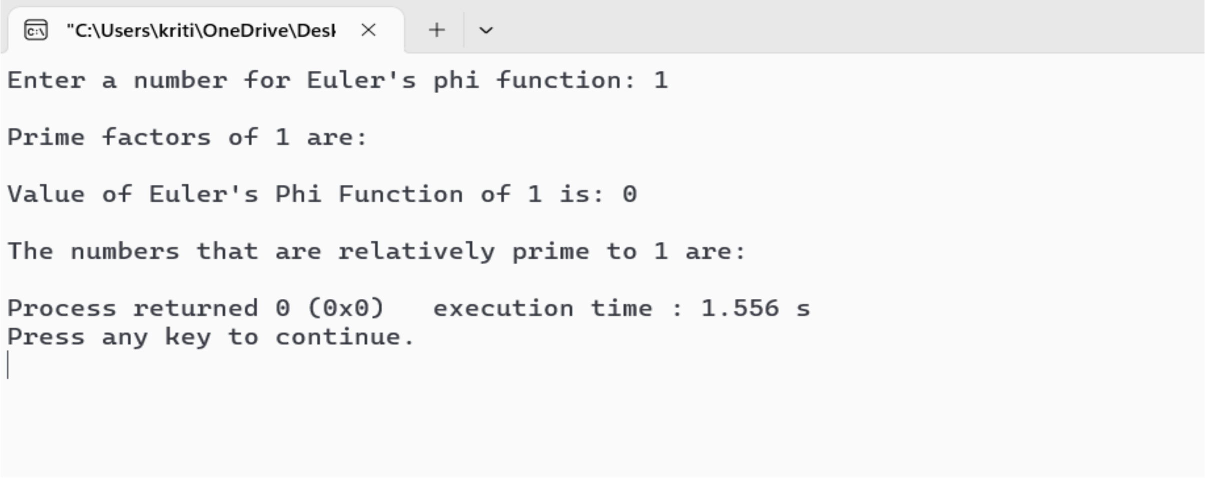
```
Enter a number for Euler's phi function: 13

Prime factors of 13 are: 13^1

Value of Euler's Phi Function of 13 is: 12

The numbers that are relatively prime to 13 are: 1  2  3  4  5  6  7  8  9  10  11  12

Process returned 0 (0x0)   execution time : 2.089 s
Press any key to continue.
```



```
Enter a number for Euler's phi function: 49

Prime factors of 49 are: 7^2

Value of Euler's Phi Function of 49 is: 42

The numbers that are relatively prime to 49 are: 1  2  3  4  5  6  8  9  10  11  12  13  15  16  17  18  19  20  22  23
 24  25  26  27  29  30  31  32  33  34  36  37  38  39  40  41  43  44  45  46  47  48

Process returned 0 (0x0)   execution time : 3.622 s
Press any key to continue.
```



```
Enter a number for Euler's phi function: 100

Prime factors of 100 are: 2^2 5^2

Value of Euler's Phi Function of 100 is: 40

The numbers that are relatively prime to 100 are: 1  3  7  9  11  13  17  19  21  23  27  29  31  33  37  39  41  43  47
  49  51  53  57  59  61  63  67  69  71  73  77  79  81  83  87  89  91  93  97  99

Process returned 0 (0x0)   execution time : 1.965 s
Press any key to continue.
```



```
Enter a number for Euler's phi function: 1

Prime factors of 1 are:

Value of Euler's Phi Function of 1 is: 0

The numbers that are relatively prime to 1 are:

Process returned 0 (0x0)   execution time : 1.556 s
Press any key to continue.
```