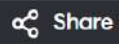


See how a CS professor is using our compiler for class assignment. [Try Programiz PRO for Educators!](#)

Programiz C Online Compiler

Programiz PRO &gt;

main.c



Run

Output

Clear

```
1 #include <stdio.h>
2 int gcdExtended(int a, int b, int* x, int* y) {
3     if (a == 0) {
4         *x = 0;
5         *y = 1;
6         return b;
7     }
8     int x1, y1;
9     int gcd = gcdExtended(b % a, a, &x1, &y1);
10    *x = y1 - (b / a) * x1;
11    *y = x1;
12    return gcd;
13 }
14 int modInverse(int e, int phi) {
15     int x, y;
16     int g = gcdExtended(e, phi, &x, &y);
17     if (g != 1)
18         return -1;
19     else
20         return (x % phi + phi) % phi;
21 }
22 int main() {
23     int e = 31;
24     int n = 3599;
```

```
Public Key (e, n): (31, 3599)
p = 59, q = 61
φ(n) = 3480
Private Key d = 3031
```

```
=== Code Execution Successful ===
```

See how a CS professor is using our compiler for class assignment. [Try Programiz PRO for Educators!](#)

main.c



Run

Output

Clear

```
11  *y = x1;
12  return gcd;
13  }
14  int modInverse(int e, int phi) {
15      int x, y;
16      int g = gcdExtended(e, phi, &x, &y);
17      if (g != 1)
18          return -1;
19      else
20          return (x % phi + phi) % phi;
21  }
22  int main() {
23      int e = 31;
24      int n = 3599;
25      int p = 59, q = 61;
26      int phi = (p - 1) * (q - 1);
27      int d = modInverse(e, phi);
28      printf("Public Key (e, n): (%d, %d)\n", e, n);
29      printf("p = %d, q = %d\n", p, q);
30      printf("φ(n) = %d\n", phi);
31      printf("Private Key d = %d\n", d);
32      return 0;
33  }
34  
```

```
Public Key (e, n): (31, 3599)
p = 59, q = 61
φ(n) = 3480
Private Key d = 3031
```

```
=== Code Execution Successful ===
```