main.c

Run

Output

Clear

```c
#include <stdio.h>
#include <stdlib.h>

// Function to compute gcd
unsigned long long gcd(unsigned long long a,
    unsigned long long b) {
    while (b != 0) {
        unsigned long long temp = b;
        b = a % b;
        a = temp;
    }
    return a;
}

// Function to compute modular inverse
unsigned long long modinv(unsigned long long a,
    unsigned long long m) {
    long long m0 = m, t, q;
```

```
Public key: (e = 17, n = 3233)
Private key: (d = 2753, n = 3233)
Encrypted message: 2557
Decrypted message: 42

--- Simulating Private Key Leak ---
Leaked private key: d = 2753
Floating point exception


=== Code Exited With Errors ===
```

main.c

Run

Output

Clear

```
79
80      // Bob tries to generate a new public
            /private key with same n
81      unsigned long long new_e = 5;
82      unsigned long long new_d = modinv(new_e, phi
            );
83
84      printf("New public key: (e = %llu, n = %llu
            )\n", new_e, n);
85      printf("New private key: (d = %llu, n = %llu
            )\n", new_d, n);
86
87      printf("\n-> Not secure! Because phi(n) is
            known from the original leaked key.\n");
88
89      return 0;
90  }
91
```

```
Public key: (e = 17, n = 3233)
Private key: (d = 2753, n = 3233)
Encrypted message: 2557
Decrypted message: 42

--- Simulating Private Key Leak ---
Leaked private key: d = 2753
Floating point exception


=== Code Exited With Errors ===
```