

SurajSG23 /  
Cryptography-Assignment

&lt;&gt; Code

Issues

Pull requests

Actions

Projects

Security

Insights

## Cryptography-Assignment / Extended Euclidean Algorithm



SurajSG23 Create Extended Euclidean Algorithm

b107d92 · 9 hours ago



66 lines (50 loc) · 1.77 KB

Code

Blame



Raw



```
1  import java.util.Scanner;
2
3  public class ExtendedEuclidean {
4
5      // Extended Euclidean Algorithm to find gcd and coefficients x, y
6      public static int[] extendedEuclid(int a, int m) {
7          if (m == 0) {
8              return new int[] { a, 1, 0 };
9          }
10
11         int[] result = extendedEuclid(m, a % m);
12         int gcd = result[0];
13         int x1 = result[1];
14         int y1 = result[2];
15
16         // Update x and y using the recursive results
17         int x = y1;
18         int y = x1 - (a / m) * y1;
19
20         return new int[] { gcd, x, y };
21     }
22
23     // Function to find the modular inverse of a mod m
24     public static int modInverse(int a, int m) {
25         int[] result = extendedEuclid(a, m);
26         int gcd = result[0];
27         int x = result[1];
28
29         // If gcd(a, m) is not 1, then the inverse does not exist
30         if (gcd != 1) {
31             return -1; // No modular inverse exists
32         } else {
33             // Ensure x is positive and return it as the modular inverse
34             return (x % m + m) % m;
35         }
36     }
37 }
```

```
37
38     public static void main(String[] args) {
39         Scanner scanner = new Scanner(System.in);
40
41         // Take input from the user
42         System.out.print("Enter a number (a): ");
43         int a = scanner.nextInt();
44
45         int m = 26; // Number of letters in the alphabet
46
47         // Calculate modular inverse
48         int inverse = modInverse(a, m);
49
50         if (inverse == -1) {
51             System.out.println("No modular inverse exists for " + a + " modulo "
52         } else {
53             System.out.println("The modular inverse of " + a + " modulo " + m +
54         }
55
56         scanner.close();
57     }
58 }
59
60 //Output
61
62 Enter a number (a): 5
63 The modular inverse of 5 modulo 26 is: 21
64
65 Enter a number (a): 13
66 No modular inverse exists for 13 modulo 26
```