### Assignment no 9: Chinese Remainder Theorem

**2020BTECS00085**

**MANJIRI CHANDURE**

**BATCH : B5**

The Chinese Remainder Theorem (CRT) is a fundamental theorem in number theory and modular arithmetic. It provides a way to solve systems of modular congruences. The theorem is named after its Chinese origins and is widely used in various mathematical and computational applications.

Here two arrays num[0..k-1] and rem[0..k-1]. In num[0..k-1], every pair is coprime (gcd for every pair is 1). Here need to find minimum positive number x such that:

x % num[0] = rem[0],

x % num[1] = rem[1],

.......................

x % num[k-1] = rem[k-1]

//

x ≡ a₁ (mod m₁)

x ≡ a₂ (mod m₂)

...

x ≡ aₖ (mod mₖ)

//

The moduli m₁, m₂, ..., mₖ must be pairwise coprime (i.e., their greatest common divisors are 1). This is a crucial condition for the theorem to work.

x ≡ (a₁ \* M₁ \* y₁ + a₂ \* M₂ \* y₂ + ... + aₖ \* Mₖ \* yₖ) (mod M)

where:

• M₁, M₂, ..., Mₖ are the moduli without the corresponding mᵢ.

• y₁, y₂, ..., yₖ are the modular inverses of M₁, M₂, ..., Mₖ with respect to m₁, m₂, ..., mₖ, respectively.

**Code:**

import java.util.Scanner;

public class CRT {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of congruences: ");

        int numCongruences = scanner.nextInt();

        int[] a = new int[numCongruences];

        int[] n = new int[numCongruences];

        for (int i = 0; i < numCongruences; i++) {

            System.out.print("Enter a\_" + (i + 1) + ": ");

            a[i] = scanner.nextInt();

            System.out.print("Enter n\_" + (i + 1) + ": ");

            n[i] = scanner.nextInt();

        }

        int x = chineseRemainderTheorem(a, n);

        System.out.println("The solution is x = " + x);

    }

    // Function to calculate the greatest common divisor (GCD) of two numbers

    public static int gcd(int a, int b) {

        if (b == 0) {

            return a;

        }

        return gcd(b, a % b);

    }

    // Function to calculate the modular multiplicative inverse using the extended Euclidean algorithm

    public static int modInverse(int a, int m) {

        for (int x = 1; x < m; x++) {

            if ((a \* x) % m == 1) {

                return x;

            }

        }

        return 1;

    }

    // Function to apply the Chinese Remainder Theorem

    public static int chineseRemainderTheorem(int[] a, int[] n) {

        int N = 1;

        for (int i = 0; i < n.length; i++) {

            N \*= n[i];

        }

        int result = 0;

        for (int i = 0; i < a.length; i++) {

            int ni = N / n[i];

            int xi = modInverse(ni, n[i]);

            result += a[i] \* ni \* xi;

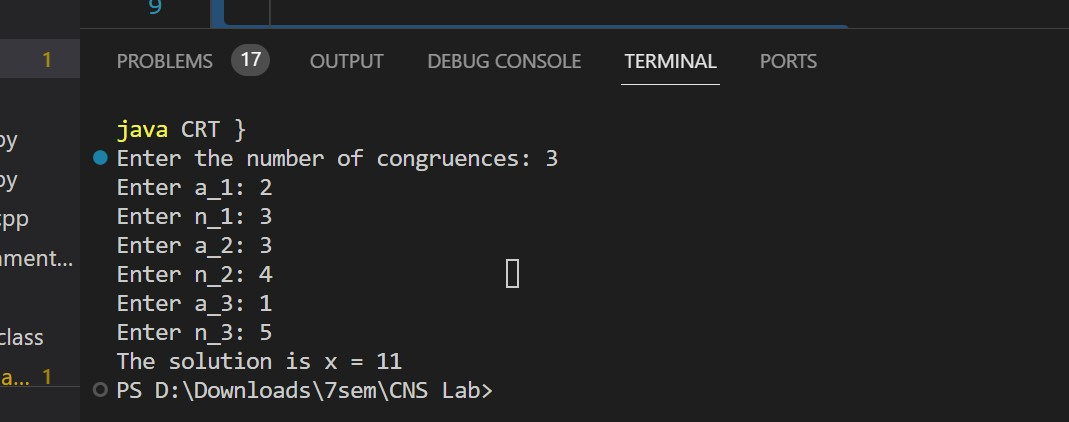
        }

        return result % N;

    }

}

**OutPut:**



**Applications:**

Cryptography: The CRT is used in cryptographic algorithms like RSA to speed up modular exponentiation.

Error Detection and Correction: It's used in error-correcting codes to correct errors in data transmission.

Computer Graphics: CRT can be applied to color representation in computer graphics.

Number Theory: CRT plays a significant role in number theory research and proofs.