**Name:** Aditya Somani **Roll No:** BE1851061 **PRN:** 71901204L

**Assignment Number: 2**

**Assignment Title:** Develop and program in C++ or Java based on number theory such as Chinese remainder.

## Relative Prime Numbers

Two integers are termed relative prime if the only common factor between them is 1.

i.e. Greatest Common Divisor(m, n) = 1

Any integer can be broken down into certain multiples of prime numbers. This is called prime factorization. When two integers are prime factorized and the only common number is 1, then the two integers are relative prime.

Two distinct primes and are always relatively prime. Relative primality is not transitive.

Example

18 = 2 x 3 x 3

35 = 7 x 5

so 18 and 35 are relative primes.

18 = 2 x 3 x 3

21 = 3 x 7

3 is common, so 18 and 21 are not relative prime.

## Set of Residues

It is a set of nonnegative integers less than n.

## Zn = {0,1,2,…….(n – 1) }

**Chinese Remainder Theorem (CRT)**

## Steps in Chinese Remainder Theorem

1. Find M = m1 × m2 × … × mk. This is the common modulus.
2. Find M1 = M/m1, M2 = M/m2… Mk = M/mk.
3. Find the multiplicative inverse of M1, M2, …, Mk using the corresponding moduli

(m1, m2, …, mk). Call the inverses M1−1, M2−1, …, Mk −1.

1. The solution to the simultaneous equations is



**Code:**

import java.io.\*;

class ChineseRemainder {

static int findMinX(int num[], int rem[], int k) {

int x = 1; // Initialize result

while (true) {

// Check if remainder of x % num[j] is rem[j] or not (for all j from 0 to k-1)

int j;

for (j = 0; j < k; j++) {

if (x % num[j] != rem[j]) {

break;

}

}

// If all remainders matched, we found x

if (j == k) {

return x;

}

// Else try next number

x++;

}

}

}

class ChineseRemainderDemo {

public static void main(String args[]) {

int num[] = {3, 4, 5};

int rem[] = {2, 3, 1};

int k = num.length;

int x = ChineseRemainder.findMinX(num, rem, k);

System.out.println("Value of x = " + x);

}

}

**Output:**

Value of x = 11