**//Chinise remainder Theorem**

#include<iostream>

using namespace std;

// Returns modulo inverse of a with respect to m using extended

int inv(int e, int r)

{

int d,b;

for(int i=1;i<5000;i++)

{

b = ((r\*i)+1);

if(b % e == 0)

break;

}

d = b/e;

return d;

}

// (gcd for every pair is 1)

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

{

// Compute product of all numbers

int prod = 1;

for (int i = 0; i < k; i++)

prod \*= num[i];

// Initialize result

int result = 0;

for (int i = 0; i < k; i++)

{

int pp = prod / num[i];

result += rem[i] \* inv(pp, num[i]) \* pp;

}

return result % prod;

}

int main(void)

{

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

int rem[] = {2,3,4};

int k = sizeof(num)/sizeof(num[0]);

cout << "x is " << findMinX(num, rem, k);

return 0;

}