Modular Arithmetic

Daniel López Montero

November 2, 2019

Contents

1 Basics

$a \equiv b(mod\, m) \leftrightarrow a - b = \lambda m, \lambda \in \mathbb{Z}$

1. Reflexibity
2. Simmetry
3. Transivity

if $a_1 \equiv b_1(mod\, m)$,$a_2 \equiv b_2(mod\, m)$,$a_3 \equiv b_3(mod\, m)$ then:

- $a + k \equiv b + k(mod\, m)$
- $ak \equiv bk(mod\, m)$
- $a_1 + a_2 \equiv b_1 + b_2(mod\, m)$
- $a_1 a_2 \equiv b_1 b_2(mod\, m)$
- $a^k \equiv b^k(mod\, m)$
- $a + k \equiv b + k(mod\, m) \rightarrow a \equiv b(mod\, m)$
- $ak \equiv bk(mod\, m) \wedge gcd(k, m) = 1 \rightarrow a \equiv b(mod\, m)$
- $a^{-1} \leftrightarrow gcd(a, m) = 1$
- $a^{-1} \equiv b^{-1}(mod\, m)$
- $ax \equiv b(mod\, m) \rightarrow x = a^{-1}b(mod\, m)$
- $a^{\phi(m)} \equiv 1(mod\, m)$
- $(p - 1)! \equiv -1(mod\, m)$

2 Great Common Divisor (GCD)

Euclides Algorithm is used to solve this problem.

$$a = bq + r$$

$$gcd(a, b) = gcd(b, r) = \ldots = gcd(c, 1) = c$$

3 Least Common Multiple (LCM)

We can calculate the LCM using GCD:

$$lcm(a, b) = ab/gcd(a, b)$$

4 Bezout's Theorem

Using Euclides Algorithm we are able to find $u and v$, where:

$$gcd(a, b) = au + bv : u, v \in \mathbb{Z}$$

5 Number Decomposition

This algorithm could be optimized because we only need to look numbers below $\sqrt{n}$, and thus, the time is reduced.

6 $a^b (mod\, m)$

we are not always to compute high numbers with big exponents, and hence we need another way that simplifies:

we write b in binary, such as: $\alpha_1 \alpha_2 \ldots \alpha_n$, where $\alpha_i$is 0 or 1.

$a^b (mod\, m) = a^{\alpha_1} \ldots a^{\alpha_n} (mod\, m)$ and $a^{\alpha_{n+1}} (mod\, m) = (a^{\alpha_n})^2 (mod\, m)$

7 Primality Test

This algorithm tells you, whether a number is prime or not (optimized).

8 Find next Prime

This function is important in cryptography. It uses an initial number and finds the next prime over the number given.

9 Euler's Totient Function

We need the number decomposition algorithm mentioned before.

$$\phi(n) = (p_1 - 1) p_1^{t_1 - 1} * \cdots * (p_n - 1) p_n^{t_n - 1}$$

10 Linear Equation Solver

$$ax \equiv b (mod\, m), b | gcd(a, m)$$

To solve the equation, we divide by $d := gcd(a, m)$

$$a' dx \equiv b' d (mod\, m' d) \rightarrow a' x \equiv b' (mod\, m')$$

We use Bezout:

$$a'u + m'v = 1$$

Then, we multiply by b':

$$b' = a'bu + m'vb \equiv a'du(mod\, m)$$

$$x = du$$

$$x_0 = du + 0m', x_1 = du + 1m', \ldots, x_{d-1} = du + (d-1)m'$$

There are $d$solutions for this equation

11 Inverse

The inverse of a number in mod m is calculated using the linear solver:
$a^{-1}x \equiv 1(mod\, m)$

12 System of Equations

Solving a system of equations is done by constantly solving two equations.

$$\begin{cases} a_1 x \equiv b_1(\mod m_1) \\ a_2 x \equiv b_2(\mod m_2) \end{cases}$$

$$gcd(m_1, m_2) = 1 = m_1 u + m_2 v$$

$x = m_1 u x_2 + m_2 v x_1$, where $x_1$and $x_2$are the solutions for each equation. We repeat the process:

$$\begin{cases} x \equiv b(mod\, m_1 m_2) \\ a_3 \equiv b_3(mod\, m_3) \end{cases}$$