Arytmetyka modularna

Robert von Oliva

2019-10-22

naruciakk

Spis treści

Kongruencje

Definicja

Dwie liczby całkowite x_1 i x_2 są wobec siebie kongruentne modulo $m \geq 0$ jeżeli różnią się one o wielokrotność liczby m, to jest $x_2 - x_1 = km$ dla każdego $k \in \mathbb{Z}$. Innymi słowy, jeżeli dzieląc obie liczby przez m otrzymamy taką samą resztę, to są one kongruentne.

Taką relację oznaczamy $x_1 \equiv x_2 \pmod{m}$.

Własności

Zwrotność – $x_1 \equiv x_2 \pmod{n}$

Symetria – $\forall x_1, x_2, n\ x_1 \equiv x_2 \pmod{n}$ oraz $x_2 \equiv x_1 \pmod{n}$

Przechodniość – jeżeli $x_1 \equiv x_2 \pmod{n}$ oraz $x_2 \equiv x_3 \pmod{n}$ to także $x_1 \equiv x_3 \pmod{n}$

Jest to więc *relacja równoważności*

Własności

Dla $a_1 \equiv a_2 \pmod{n}$ oraz $b_1 \equiv b_2 \pmod{n}$:

translacja: $a_1 + k \equiv a_2 + k \pmod{n}$

skalowanie: $ka_1 \equiv ka_2 \pmod{n}$

dodawanie: $a_1 + b_1 \equiv a_2 + b_2 \pmod{n}$

odejmowanie: $a_1 - b_1 \equiv a_2 - b_2 \pmod{n}$

mnożenie: $a_1 b_1 \equiv a_2 b_2 \pmod{n}$

potęgowanie: $a_1^2 \equiv a_2^2 \pmod{n}$

$W(a_1) \equiv W(a_2) \pmod{n}$ dla dowolnego wielomianu $W(x)$ o współczynnikach całkowitych

Chińskie twierdzenie o resztach

Liniowa kongruencja

$ax \equiv b \pmod{n}$, gdzie x jest zmienną.

Przykłady:

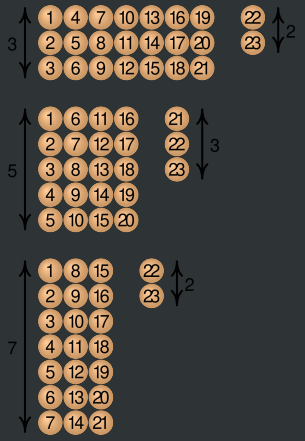
1. Liniowa kongruencja $2x \equiv 1 \pmod{3}$ ma wiele rozwiązań, jednym z nich jest $x = 2$, bo $2 \cdot 2 = 4 \equiv 1 \pmod{3}$. Zbiór wszystkich rozwiązań to: $x \in \{\ldots, 2, 5, 8, 11, \ldots\}$.
2. Kongruencja $4x \equiv 1 \pmod{2}$ nie ma rozwiązań, gdyż $4x$ zawsze jest parzyste, więc podzielone przez 2 nigdy nie da reszty 1.

Twierdzenie

Zakładając, że $m_1, m_2, \ldots, m_n$ są parami względnie pierwsze (tj. $\forall_{i \neq j}\ nwd(m_i, m_j) = 1$), a $a_1, a_2, \ldots, a_n \in \mathbb{Z}$, to układ kongruencji

$$\begin{aligned} x &\equiv a_1 \pmod{m_1} \\ x &\equiv a_2 \pmod{m_2} \\ &\vdots \\ x &\equiv a_k \pmod{m_k}, \end{aligned}$$

dla $i = 1, 2, \ldots, n$ ma dokładnie jedno rozwiązanie modulo $M = m_1 m_2 \ldots m_n$ ze względu na niewiadomą x.



Rysunek 1: Cmglee, Wikicommons, CC BY-SA 4.0

Schemat rozwiązywania układu kongruencji

Niech $M = m_1 \cdot m_2 \cdot \ldots m_k$ oraz $\forall_{i \leq k} M_i = \frac{M}{m_i}$ gdzie m_i i M_i są względnie pierwsze.

Korzystając z rozszerzonego algorytmu Euklidesa możemy wyznaczyć liczby f_i i g_i spełniające poniższe równanie:

$$\forall_{i \leq k} f_i m_i + g_i M_i = NWD(m_i, M_i) = 1$$

Zauważmy, że $g_i M_i \equiv 1 \pmod{m_i}$ oraz $g_i M_i \equiv 0 \pmod{m_j}$ dla $i \neq j$. Wtedy możemy wyliczyć x przy użyciu wzoru:

$$x = \sum_{i=1}^{k} a_i g_i M_i$$

Jest to jedno z rozwiązań, pozostałe różnią się o wielokrotność M.

System resztowy

System liczbowy służący do reprezentacji liczb całkowitych za pomocą zbioru (wektora) reszt z dzielenia tej liczby przez pewne ustalone liczby naturalne. Zgodnie z chińskim twierdzeniem o resztach taka reprezentacja jest jednoznaczna dla liczb całkowitych ze zbioru $[0, m_1 m_2 m_3 \dots m_n)$.

Działania:

1. dodawanie: $C = A + B \pmod{M}$, $c_i = a_i + b_i \pmod{m_i}$
2. odejmowanie: $C = A - B \pmod{M}$, $c_i = a_i - b_i \pmod{m_i}$
3. mnożenie: $C = A \cdot B \pmod{M}$, $c_i = a_i \cdot b_i \pmod{m_i}$
4. dzielenie: *it's a bit more complicated*

Konwersja

Na system resztowy:

$A = 42159_{10}$
Baza systemu resztowego: 15, 16, 19
$42159 \equiv 9 \pmod{15}$
$42159 \equiv 15 \pmod{16}$
$42159 \equiv 17 \pmod{19}$

$42159_{10} = (9, 15, 17)_{(15,16,19)}$

Z systemu resztowego:

m_i – kolejne elementy bazy
$A = [a_1, a_2, \ldots, a_n]$
Szukamy takich liczb $k_1, k_2, \ldots, k_i \in \mathbb{N}$ dla których:

$$\forall_{i \in \mathbb{N}, i \leq n} m_0 \cdot \ldots \cdot m_{i-1} \cdot k_i \cdot m_{i+1} \cdot \ldots \cdot m_N \pmod{m_i} = 1$$

Obliczywszy to, liczbę możemy uzyskać z równania:

$$A = \sum_{i=1}^{n} m_0 \cdot \ldots \cdot m_{i-1} \cdot k_i \cdot a_i \cdot m_{i+1} \cdot \ldots \cdot m_N \pmod{\prod_{i=1}^{n} m_i}$$

Twierdzenia Fermata i Eulera

Jak zachowują się reszty z dzielenia?

Z własności mnożenia kongruencji wiemy, że dla $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$ zachodzi $ac \equiv bd \pmod{m}$. Mając więc kilka pierwszych wyników dzielenia modulo:

$7^1 \equiv 3 \pmod{4}$
$7^2 \equiv 1 \pmod{4}$
$7^3 \equiv 3 \pmod{4}$
$7^4 \equiv 1 \pmod{4}$

Dla 7^2 kongruencja wynosi 1, więc startując odpowiednio od 7^1 i 7^2 możemy udowodnić, że dla wszystkich parzystych wykładników kongruencja wynosi 1, a dla wszystkich nieparzystych 3.

$4^1 \equiv 4 \pmod{15}$
$4^2 \equiv 1 \pmod{15}$
$4^3 \equiv 4 \pmod{15}$
$4^4 \equiv 1 \pmod{15}$

$5^1 \equiv 5 \pmod{13}$
$5^2 \equiv 12 \pmod{13}$
$5^3 \equiv 8 \pmod{13}$
$5^4 \equiv 1 \pmod{13}$
$5^5 \equiv 5 \pmod{13}$
$5^6 \equiv 12 \pmod{13}$

Maksymalna długość ciągów

Dla każdego ciągu $a^1, a^2, a^3, \ldots \pmod{m}$ jesteśmy w stanie określić pewnien powtarzający się wzorzec reszt.

1. Z m możliwych wyników 0 występuje wyłącznie w ciągach, które są zerowe, gdyż dla $a^n \equiv 0 \pmod{m}$ możemy stwierdzić, że $a^{n+1} \equiv 0 \pmod{m}$, gdyż dla $a \equiv k \pmod{n}, k \in \mathbb{N}$ po przemnożeniu przez $a^n \equiv 0 \pmod{m}$ otrzymujemy $a^n \cdot a \equiv 0 \cdot k \pmod{m}$ czyli $a^{n+1} \equiv 0 \pmod{m}$.
2. Żadna z cyfr nie może powtórzyć się w ciągu, gdyż nie może mieć dwóch różnych następników. Każdy następny element ciągu da się określić poprzez przemnożenie go przez kongruencję a^1.

Małe twierdzenie Fermata

Jeżeli p to liczba pierwsza, to dla dowolnej liczby całkowitej a, liczba $a^p - a$ jest podzielna przez p.

Innymi słowy, jeżeli p jest liczbą pierwszą, a a jest liczbą całkowitą dla której p jest względnie pierwsze, to $a^{p-1} - 1 \equiv 0 \pmod{p}$, czyli $a^{p-1} \equiv 1 \pmod{p}$.

Test pierwszości Fermata: wybieramy kilka losowych wartości a względnie pierwszych z p, a nastepnie sprawdzamy, czy Małe twierdzenie Fermata jest dla nich spełnione. Jeżeli dla którejkolwiek nie jest to jest to liczba złożona, jeżeli wszystkie spełniają to p prawdopodobnie jest liczbą pierwszą, albo pseudopierwszą.

Funkcja Eulera

Określa ile jest liczb względnie pierwszych mniejszych lub równych podanej liczbie n, tj. znajduje dla niej liczby względnie pierwsze. Taką funkcję oznaczamy $\phi(n)$.

Podstawowe własności:

1. $\phi(p) = p - 1$, gdzie p jest liczbą pierwszą
2. $\phi(mn) = \phi(m)\phi(n)$, gdzie m, n są względnie pierwsze
3. $\phi(p^k) = p^{k-1} \cdot (p - 1)$, gdzie p jest liczbą pierwszą

Funkcja Eulera

Istnieją także inne, równie ciekawe własności funkcji Eulera:

1. $\phi(mn) = \phi(m)\phi(n) \cdot \frac{NWD(m,n)}{\phi(NWD(m,n))}$
2. $\phi(NWW(m,n)) \cdot \phi(NWW(m,n)) = \phi(m)\phi(n)$
3. $\frac{\phi(n)}{n} = \frac{\phi(rad(n))}{rad(n)}$, gdzie $rad(n) = \prod_{p|n} p$, p jest liczbą pierwszą

Twierdzenie Eulera

Dwie liczby $m \in \mathbb{N}$ i $a \in \mathbb{Z}$ spełniają następującą kongruencję:

$$a^{\phi(m)} \equiv 1 \pmod{m}$$

gdzie $\phi(m)$ to wartość funkcji Eulera dla m, a liczby a i m są względnie pierwsze.

Przykład: weźmy liczbę $7^{222} \pmod{10}$. Liczby $a = 7$ i $m = 10$ są względnie pierwsze, zaś wartość funkcji Eulera $\phi(10) = 4$. Redukując pierwotną potęgę otrzymujemy $7^{222} \equiv 7^{4 \cdot 55 + 2} \equiv (7^4)^{55} \cdot 7^2 \equiv 1^{55} \cdot 7^2 \equiv 49 \equiv 9 \pmod{10}$.

Małe twierdzenie Fermata – dowód

Twierdzenie: Jeżeli p jest liczbą pierwszą, a a jest liczbą całkowitą dla której p jest względnie pierwsze, to $a^{p-1} - 1 \equiv 0 \pmod{p}$, czyli $a^{p-1} \equiv 1 \pmod{p}$.

Zakładając, że twierdzenie Eulera jest poprawne, pamiętając, że p i a są względnie pierwsze, możemy opisać małe twierdzenie Fermata jako szczególny przypadek twierdzenia Eulera, gdyż $\phi(p) = p - 1$ dla liczby pierwszej p.

□

Twierdzenie Eulera – dowód

1. Weźmy dwie liczby względnie pierwsze, $a \in \mathbb{Z}$ i $m \in \mathbb{N}$ oraz ciąg liczb $a_1, a_2, \ldots, a_{\phi(m)}$.
2. Weźmy ciąg $aa_1, aa_2, \ldots, aa_{\phi(m)}$. Dla każdego i, aa_i jest również względnie pierwsze z m, więc $aa_i \equiv a_k$ dla jakiegoś k. Skoro jednak $aa_i \equiv aa_j \leftrightarrow a_i \equiv a_j \pmod{m}$, każde a_i zostanie przekształcone w inne a_k, więc jest to permutacja ciągu z punktu pierwszego.
3. $a_1 a_2 \ldots a_{\phi(m)} \equiv (aa_1)(aa_2)\ldots(aa_{\phi(m)})$
4. $a_1 a_2 \ldots a_{\phi(m)} \equiv a^{\phi(m)} a_1 a_2 \ldots a_{\phi(m)}$
5. $1 \equiv a_{\phi(m)}$

□

Logarytm dyskretny

Definicja

Weźmy dowolną grupę G z operacją mnożenia dla której elementem neutralnym jest 1. Dla dowolnego elementu grupy $b \in G$ możemy zdefiniować potęgowanie do potęgi $k \in \mathbb{N}$:

$$b^k = \underbrace{b \cdot b \cdots b}_{k}, \text{ dla } k = 0 \text{ mamy } b^0 = 1.$$

Takie k, które spełnia równanie $b^k = a$, $a \in G$ nazywamy **logarytmem dyskretnym** i zapisujemy $k = log_b a$.

tl;dl: Logarytm, który przyjmuje wyłącznie wartości naturalne.

Dlaczego jest ciekawy?

Nie istnieje obecnie algorytm pozwalający na obliczenie w czasie wielomianowym.

Algorytmy klasyczne osiągają co najwyżej złożoność $\sqrt{n}$ dla n będącego liczbą elementów grupy G, jest algorytmem wykładniczym dla liczby cyfr w tej liczbie.

Można rozwiązać ten problem algorytmem Shora.

Zastosowania

Proste sumy kontrolne

1. **IBAN** – dwie pierwsze cyfry numeru konta to suma kontrolna
   PL**62** 1160 2202 0000 0003 1280 6258
   116022020000000312806258PL**62**
   PL $\rightarrow$ 2521
   116022020000000312806258252**162** $\equiv 1 \pmod{97}$
   116022020000000312806258252**100** $\equiv 36 \pmod{97}$, a $98 - 32 = 62$
2. **Karty płatnicze** – algorytm Luhna
   Dla numeru karty: 1234567890123452 (16 cyfr)
   Mnożymy kolejne cyfry karty przez wagi (2, 1, 2, 1, ...) i sumujemy
   $1 \cdot 2 + 2 \cdot 1 + 3 \cdot 2 + \ldots = 60$, jeżeli wynik modulo 10 jest równy 0 to numer karty jest poprawny: $60 \equiv 0 \pmod{10}$.

Algorytm Rivesta-Shamira-Adlemana

Generowanie kluczy

1. Wybierz dwie liczby pierwsze: p i q o podobnej długości w bitach, ale odległe wartościami
2. Oblicz $\phi(n) = (p-1)(q-1)$ dla $n = pq$
3. Wybierz liczbę e taką, że:
   3.1 jest względnie pierwsza z $\phi(n)$,
   3.2 $1 < e < \phi(n)$.
4. Znajdź liczbę d spełniającą poniższą własność

$$d \cdot e \equiv 1 \pmod{\phi(n)}$$

Kluczem publicznym jest para liczb (n, e), zaś kluczem prywatnym (n, d).

Algorytm Rivesta-Shamira-Adlemana

Szyfrowanie wiadomości

1. Podziel wiadomość na bloki $M = (m_1, m_2, \ldots)$, każdy o wartości liczbowej nie większej niż n.
2. Każdy z bloków zaszyfruj w następujący sposób:

$$s_i = m_i^e \pmod{n}$$

Deszyfrowanie wiadomości

1. Otrzymana zaszyfrowana wiadomość jest podzielona na bloku $S = (s_1, s_2, \ldots)$.
2. Każdy z bloków odszyfruj (przekształć w tekst jawny) w następujący sposób:

$$m_i = s_i^d \pmod{n}$$

Algorytm ElGamal

Generowanie kluczy

1. Wygeneruj grupę cykliczną G rzędu q z generatorem g. Niech e będzie elementem jednostkowym G.
2. Wybierz losowo wyraz a z przedziału $1, \ldots, q-1$.
3. Oblicz $k = g^a$.
4. Klucz publiczny to zbiór wartości (G, q, g, k), zaś kluczem prywatnym jest (G, q, g, k, a).

Szyfrowanie wiadomości

1. Przedstaw wiadomość M jako element m grupy G oraz wybierz losowo wyraz b z przedziału $1, \ldots, q-1$.
2. Oblicz *sekret*, tj. $s = k^b$, $c_1 = g^b$ oraz $c_2 = m \cdot s$.
3. Zaszyfrowany sekret to para liczb (c_1, c_2).

Algorytm ElGamal

Deszyfrowanie wiadomości

1. Podnieś otrzymane $c_1 = g^b$ do potęgi a:

$$(g^b)^a = g^{ab} = (g^a)^b = k^b$$

2. Znajdź odwrotność liczby k^b przy użyciu rozszerzonego algorytmu Euklidesa:

$$\alpha k^b + \beta q = 1$$

$$\alpha k^b \equiv 1 \pmod{q}$$

$$\alpha \equiv \frac{1}{k^b} \pmod{q}$$

3. Pomnóż $m \cdot k^b$ przez odwrotność k^b, czyli α

$$(m \cdot k^b) \cdot \alpha \equiv m \cdot (b \cdot \alpha) \equiv m \cdot 1 \equiv 1 \pmod{q}$$

Czas na pytania!

Bibliografia i dodatkowe materiały do poczytania i

1. Cormen T., Leiserson C., Rivest R., Stein C., Wprowadzenie do algorytmów, Wydawnictwo Naukowe PWN, Warszawa 2013
2. Joux A., Odlyzko A., Pierrot C., The Past, evolving Present and Future of Discrete Logarithm, Open Problems in Mathematics, 2004
3. Simpson H., Modular Arithmetic Hashing Pseudorandom Sequences, 2003
4. Raji W., An Introductory Course in Elementary Number Theory, Saylor Foundation, 2013
5. Cynk S., Kongruencje, 2003
6. Hefferon J., Clark W., Elementary Number Theory, University of South Florida, Tampa, 2003

Bibliografia i dodatkowe materiały do poczytania ii

7. Zhang Y., Ge R., COMPSCI 330: Design and Analysis of Algorithms, Hashing, 2017
8. Lazar E., Number Theory II: Modular Arithmetic, Cryptography, and Randomness, 2016
9. Preneel B., Hash Functions Based on Block Ciphers and Modular Arithmetic, 2016
10. Bajard J., Meloni N., Plantard T., Efficient RNS Bases for Cryptography, 2006
11. Biernat J., Systemy resztowe, 2009
12. Sato N., Number theory