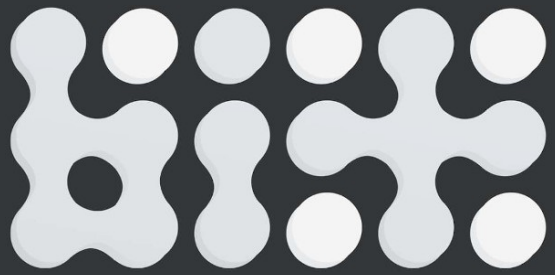

Algebra Komputerowa

Algorytmy Modularne [1, 2]

Filip Zieliński

2025

Spis Treści

KOŁO NAUKOWE

Problem

Rozważmy układ kongruencji

$$\begin{cases} a \equiv 1 \pmod 5 \\ a \equiv 6 \pmod 7 \\ a \equiv 5 \pmod 9 \end{cases}$$

pojawiają się naturalne pytania

- Czy układ ma rozwiązanie i czy da się je znaleźć?
- Jeśli tak, to ile jest tych rozwiązań?
- Jak znaleźć wszystkie rozwiązania?

Chińskie Twierdzenie o Resztach (CRT)

Niech R będzie pierścieniem Euklidesowym oraz $m_1, \ldots, m_k, a_1, \ldots a_k \in R$. Oznaczmy $M = \prod_{i=1}^{k} m_i$. Jeżeli $m_1, \ldots, m_k$ są parami względnie pierwsze to układ

$$\begin{cases} a \equiv a_1 & (\text{mod } m_1) \\ a \equiv a_2 & (\text{mod } m_2) \\ \vdots & \\ a \equiv a_k & (\text{mod } m_k) \end{cases}$$

ma dokładnie jedno rozwiązanie $\in R/\langle M \rangle$ (dokładnie jedno rozwiązenie modulo M.)

CRT - Wersja Zaawansowana

Chińskie Twierdzenie o Resztach

Chińskie Twierdzenie o Resztach (CRT)

Niech R będzie pierścieniem oraz $I_1, \ldots I_k$ parami względnie pierwszymi ideałami w tym pierścienu. Wtedy

$$R/(I_1 \cap \ldots \cap I_k) \cong R/I_1 \times \ldots \times R/I_k.$$

Uwaga

Oba prezentowane algorytmy, bazują na możliwości wyznaczenia współczynników Bezouta z Rozszerzonego Algorytmu Euklidesa, zatem wprost odnoszą się tylko do pierścieni Euklidesowych (standardowo dla nas $\mathbb{Z}, k[x]$)

Założenia: pierścień euklidesowy R.
Wejście:

- $a_1, \ldots, a_k \in R$
- $m_1, \ldots, m_k \in R$

Wyjście:

- $a : a \equiv a_i \pmod{m_i}$ dla każdego $i = 1, \ldots, k$.

Kroki:

1. Dla każdego $i \in \{1, \ldots k\}, j \in \{i+1, \ldots, k\}$ wyznacz α_{ij}, α_{ji} takie, że $\alpha_{ij} m_i + \alpha_{ji} m_j = 1$ (współczynniki Bezouta).
2. Zdefiniuj $M := \prod_{i=1}^{k} m_i$.
3. Dla każdego $i \in \{1, \ldots, k\}$ wyznacz $A_i := \frac{M}{m} \prod_{j=1, j \neq i}^{k} \alpha_{ji}$.
4. Zwróć $a := \sum_{i=1}^{k} a_i A_i \pmod{M}$.

Wady Algorytmu Lagrange'a

Układy Kongruencji

KOŁO NAUKOWE

Uwaga

Współczynniki A_i z poprzedniego algorytmu bardzo szybko rosną oraz nie jest to algorytm "przyrostowy"

Wejście::

- $a_1, a_2 \in R$
- $m_1, m_2 \in R$

Wyjście:

- a taki, że $a \equiv a_1 \pmod{m_1} \wedge a \equiv a_2 \pmod{m_2}$

Kroki:

1. Wyznacz α, β takie, że $\alpha m_1 + \beta m_2 = 1$.
2. Zdefiniuj $r := mod(a_1, m_1)$
3. Wyznacz $b := (a_2 - r)\alpha$
4. Wyznacz $s := b \pmod{m_2}$
5. Zwróć $r + sm_1$

Wejście:

- $a_1, \ldots, a_k \in R$
- $m_1, \ldots, m_k \in R$

Wyjście:

- $a : a \equiv a_i \pmod{m_i}$ dla każdego $i = 1, \ldots, k$.

Kroki:

1. Zdefiniuj $M := 1, A := 1$
2. dla każdego $i \in \{1, \ldots, k\}$
   - Używając poprzedniego algorytmu, znajdź b takie, że $b \equiv A \pmod{M} \wedge b \equiv a_i \pmod{m_i}$.
   - Zamień $A := b$ oraz $M := M \cdot m[i]$.
3. Zwróć A.

Szybkie potęgowanie modulo

KOŁO NAUKOWE

Wejście:

- $a \in \mathbb{Z}, n, m \in \mathbb{N}$

Wyjście:

- $a^n \pmod m$

Kroki:

1. Zdefiniuj $(d_k \dots d_0)_2$ jako binarny zapis wykładnika n.
2. Zainicjalizuj $r := 1, b := a \pmod m$
3. Dla każdego $i \in \{1, \dots, k\}$
4. - Jeżeli $d_i = 1$ to podstaw $r := r \cdot b \pmod m$
   - Podstaw $b := b^2 \pmod m$
5. Zwróć r

Twierdzenie (Fermat, 1640)

Jeżeli p jest liczbą pierwszą, to dla każdego $a \in \mathbb{Z}$ zachodzi

$$a^{p-1} \equiv \begin{cases} 0 \pmod{p}, & p \mid a \\ 1 \pmod{p}, & p \nmid a \end{cases}$$

Twierdzenie (Fermat, 1640)

Jeżeli p jest liczbą pierwszą, to dla każdego $a \in \mathbb{Z}$ zachodzi

$$a^{p-1} \equiv \begin{cases} 0 \pmod{p}, & p \mid a \\ 1 \pmod{p}, & p \nmid a \end{cases}$$

Obserwacja

To twierdzenie, pozwala często przyspieszyć potęgowanie modularne (zmniejszyć błyskawicznie wykładnik). Niestety tylko i wyłącznie, gdy potęgujemy modulo liczba pierwsza.

Definicja

Funkcją Eulera nazywamy funkcję $\varphi : \mathbb{N} \to \mathbb{N}$ zdefiniowaną jako

$$\varphi(n) = \#\{a \in \{0, \ldots, n-1\} \mid NWD(a, n) = 1\}$$

Innymi słowami $\varphi(n)$ to liczba jedności w pierścieniu ilorazowym $\mathbb{Z}/\langle n \rangle$.

Definicja

Funkcją Eulera nazywamy funkcję $\varphi : \mathbb{N} \to \mathbb{N}$ zdefiniowaną jako

$$\varphi(n) = \#\{a \in \{0, \ldots, n-1\} \mid NWD(a, n) = 1\}$$

Innymi słowami $\varphi(n)$ to liczba jedności w pierścieniu ilorazowym $\mathbb{Z}/\langle n \rangle$.

Obserwacja

Jeżeli p jest liczba pierwszą, to $\varphi(p) = p - 1$

Twierdzenie

- Jeżeli p jest liczbą pierwszą, to $\varphi(p^k) = p^k - p^{k-1}$
- Jeżeli $\mathrm{NWD}(\mathrm{m}, \mathrm{n}) = 1$ to $\varphi(mn) = \varphi(m) \cdot \varphi(n)$

Twierdzenie (Euler, 1736)

Niech $m > 1$ będzie liczbą całkowitą dodatnią. Jeżeli $a \in \mathbb{Z}$ jest względnie pierwsze z m to zachodzi

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

Twierdzenie (Euler, 1736)

Niech $m > 1$ będzie liczbą całkowitą dodatnią. Jeżeli $a \in \mathbb{Z}$ jest względnie pierwsze z m to zachodzi

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

Obserwacja

To twierdzenie również bardzo przyspiesza potęgowanie modularne.

Obserwacja

Rozważmy macierz kwadratową $M \in \mathcal{M}_n(\mathbb{Z})$. W oczywisty sposób $\det M \in \mathbb{Z}$.
Mimo tego, wykorzystując klasyczne algorytmy liczenia wyznacznika (eliminacja Gaussa) na komputerze z powodu dzielenia możemy natrafić na liczby zmiennoprzecinkowe i końcowy wynik może nie być dokładny.
Pomysł: W ciałach skończonych $\mathbb{Z}_p$ operację są dokładne - można policzyć wyznacznik modulo wiele liczb pierwszych, a później z Chińskiego Twierdzenia o Resztach złożyć wiele wyników w jeden ostateczny. Dodatkowa zaleta: Operacje modulo są stosunkowo szybkie!

Twierdzenie

Niech $M = (m_{ij}) \in \mathcal{M}_n(\mathbb{Z})$ będzie macierza kwadratową $n \times n$, a p dowolną liczbą pierwszą. Niech $\overline{m}_{ij} = (m_{ij} \pmod p)$ oraz $\overline{M} = (\overline{m}_{ij})$. Zachodzi

$$\det \overline{M} \equiv \det M \pmod p$$

Twierdzenie (Hadamard, 1893)

Niech $M = (m_{ij}) \in \mathcal{M}_n(\mathbb{R})$ będzie rzeczywistą macierzą kwadratową. Zachodzi

$$|\det M| \leqslant \prod_{j=1}^{n} \sqrt{\sum_{i=1}^{n} m_{ij}^2}$$

Twierdzenie (Hadamard, 1893)

Niech $M = (m_{ij}) \in \mathcal{M}_n(\mathbb{R})$ będzie rzeczywistą macierzą kwadratową. Zachodzi

$$|\det M| \leqslant \prod_{j=1}^{n} \sqrt{\sum_{i=1}^{n} m_{ij}^2}$$

Wniosek

Jeżeli rozważymy $B > 0$ takie, że $|m_{ij}| \leqslant B$ dla wszystkich $i, j \leqslant n$, to zachodzi

$$|\det M| \leqslant B^n \sqrt{n^n}$$

Wejście:

- $M = (m_{ij})$ macierz całkowitoliczbowa wymiaru $n \times n$.

Wyjście:

- $\det M$.

Kroki:

1. Wyznacz $B \geqslant |m_{ij}||$ dla każdego $i, j \leqslant n$.
2. Znajdź liczby pierwsze $p_1, \ldots p_k \geqslant 3$ takie, że $m := p_1 \cdots p_k > 2B^n\sqrt{n^n}$.
3. Dla każdego $i \in \{1, \ldots, k\}$ oblicz $d_i := det(M \pmod{p_i}) \in \mathbb{Z}_p$.
4. Rozwiąż układ kongruencji $d \equiv d_i \pmod{p_i}$ dla $i \in \{1, \ldots, k\}$.
5. Jeśli $d > m/2$ to podstaw $d := d - m$.
6. Zwróć d.

Prezentacja jest mocno oparta o wykład autorstwa *Przemysława Koprowskiego*, który można obejrzeć pod tym linkiem

[1] Joachim Von Zur Gathen and Jurgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999.

[2] Przemysław Koprowski. *Lectures on Computational Mathematics*. 2022.

Pytania, wątpliwości, uwagi ?