BLG501E – Discrete Mathematics

2021 – 2022 Fall Term

Asst. Prof. Gökhan SEÇİNTİ

Outline

- The Chinese Remainder Theorem (cont`d)
- Euclidian Algorithm (revisit)
- Computer Arithmetic for Large Integers
- Fibonacci Sequence
- Lame Theorem
- Wilson Theorem
- Fermat`s Little Theorem
- Pseudo Prime Numbers

The Chinese Remainder Theorem

- In the first century, the Chinese mathematician Sun-Tsu asked:

  There are certain things whose number is unknown. When divided by 3, the remainder is 2; when divided by 5, the remainder is 3; when divided by 7, the remainder is 2. What will be the number of things?

- This puzzle can be translated into the solution of the system of congruences:

  $x \equiv 2 \pmod 3$,

  $x \equiv 3 \pmod 5$,

  $x \equiv 2 \pmod 7$?

- We'll see how the theorem that is known as the *Chinese Remainder Theorem* can be used to solve Sun-Tsu's problem.

The Chinese Remainder Theorem

Theorem 2: (*The Chinese Remainder Theorem*) Let $m_1, m_2, \ldots, m_n$ be pairwise relatively prime positive integers greater than one and $a_1, a_2, \ldots, a_n$ arbitrary integers. Then the system

$$x \equiv a_1 \pmod{m_1}$$
$$x \equiv a_2 \pmod{m_2}$$
$$\vdots$$
$$x \equiv a_n \pmod{m_n}$$

has a unique solution modulo $m = m_1 m_2 \cdots m_n$.

(That is, there is a solution x with $0 \le x < m$ and all other solutions are congruent modulo m to this solution.)

- **Proof**: We'll show that a solution exists by describing a way to construct the solution. Showing that the solution is unique modulo m is Exercise 30.

continued →

The Chinese Remainder Theorem

To construct a solution first let $M_k = m/m_k$ for $k = 1,2,\ldots,n$ and $m = m_1 m_2 \cdots m_n$.

Since $\gcd(m_k, M_k) = 1$, by Theorem 1, there is an integer y_k, an inverse of M_k modulo m_k, such that

$$M_k y_k \equiv 1 \pmod{m_k}.$$

Form the sum

$$x = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_n M_n y_n .$$

Note that because $M_j \equiv 0 \pmod{m_k}$ whenever $j \neq k$, all terms except the kth term in this sum are congruent to 0 modulo m_k.

Because $M_k y_k \equiv 1 \pmod{m_k}$, we see that $x \equiv a_k M_k y_k \equiv a_k \pmod{m_k}$, for $k = 1,2,\ldots,n$.

Hence, x is a simultaneous solution to the n congruences.

$$\begin{aligned} x &\equiv a_1 \pmod{m_1} \\ x &\equiv a_2 \pmod{m_2} \\ &\ \vdots \\ x &\equiv a_n \pmod{m_n} \end{aligned}$$

◄

The Chinese Remainder Theorem

Proof:

Solution: $x = \sum_0^n a_i y_i M_i$ where $M_i = \frac{m}{m_i}$

Part 1: Showing that a solution exists

Part 2: Showing that a solution is unique for the given interval

The Chinese Remainder Theorem

Part 1: $m_1 \perp m_2 \ \dots \ \perp \mathrm{m_n}, \forall i \in [1, n] m_i | c \iff m|c$

Induction

1. $m_1 | c$
2. $\forall i \in [1, n-1] m_i | c \iff M_n | c$
3. $m_n | c \wedge M_n | c \wedge m_n \perp M_n \Rightarrow^? m|c$

$M_n | c \rightarrow c = k_1 M_n$

$m_n | c \rightarrow m_n | k_1 M_n \wedge m_n \perp M_n \rightarrow m_n | k_1 \rightarrow k_1 = k_2 m_n$

$c = k_2 m_n M_n \rightarrow m_n M_n | c \rightarrow m|c$

The Chinese Remainder Theorem

Part 2:

Contradiction, lets assume both x and y are different solutions in [1,m]

$$\forall i \in [1, n]; x \equiv a_i \, mod \, m_i \wedge y \equiv a_i \, mod \, m_i$$
$$x = m_i q_1 + a_i \wedge y = m_i q_2 + a_i \Rightarrow x - y = m_i q_3$$

$$m_i | (x - y) \Rightarrow x \equiv y \, mod \, m_i$$

$$\forall i, m_i | (x - y) \wedge m = \prod m_i \Rightarrow m | (x - y) \Rightarrow x \equiv y \, mod \, m$$

Extended Chinese Remainder Theorem

$a_1 x \equiv b_1 \ (\bmod\ m_1)$
$a_2 x \equiv b_2 \ (\bmod\ m_2)$
$\cdot$
$\cdot$
$\cdot$
$a_n x \equiv b_n \ (\bmod\ m_n)$

In order to have solution for the given system, each congruence should have solution, which requires

Either

$\gcd(a_i, m_1) = 1$

Or

$\gcd(a_i, m_1) = d_i \ \wedge \ d_i \mid b_i$

Extended Chinese Remainder Theorem

Example:

$15x \equiv 21\, mod 48$

$166x \equiv 46\, mod\ 22$, x =?

$x \equiv 5\, mod\ 13$

Euclidean Algorithm (revisit)

$$\gcd(a, b) = \gcd(b, a \bmod b)$$

- The Euclidean algorithm expressed in pseudocode is:

```
procedure gcd(a, b: positive integers)
x := a
y := b
while  y ≠ 0
    r := x mod y
    x := y
    y := r
return x {gcd(a,b) is x}
```

- In Section 5.3, we'll see that the time complexity of the algorithm is $O(\log b)$, where $a > b$.

Computer Arithmetics with Large Integers

$$m_1 \perp m_2 \ \ldots \perp \mathrm{m_n} \ \textit{and}\ m = m_1 m_2 \ \ldots \ \mathrm{m_n}$$

Any large integer $a \in [1, m-1]$ can be converted into n smaller integers

$$(a \bmod m_1, a \bmod m_2 \ \ldots, a \bmod \mathrm{m_n})$$

$X = < x_1, x_2 \ldots, x_n >$
$Y = < y_1, y_2 \ldots, y_n >$

$\Delta = \{+, -, x, \div\}$

$X \Delta Y = < x_1 \Delta y_1, x_2 \Delta y_2 \ldots, x_n \Delta y_n >$

Computer Arithmetics

Theorem:

$$\gcd\left(2^a - 1, 2^b - 1\right) = 2^{\gcd(a,b)} - 1$$

$Examples$:

$31 \perp 29 \perp 27 \perp 25 \perp 23 \perp 19$

$2^{31} - 1 \perp 2^{29} - 1 \perp 2^{27} - 1 \perp 2^{25} - 1 \perp 2^{23} \perp 2^{19} - 1$

Computer Arithmetics

Theorem:

$$\gcd\left(2^{a}-1,2^{b}-1\right)=2^{\gcd(a,b)}-1$$

Proof:

Fibonacci Numbers

Fibonacci
(1170- 1250)

Example : The Fibonacci numbers are defined as follows:

$f_0 = 0$

$f_1 = 1$

$f_n = f_{n-1} + f_{n-2}$

Find f_2, f_3, f_4, f_5 .

- $f_2 = f_1 + f_0 = 1 + 0 = 1$
- $f_3 = f_2 + f_1 = 1 + 1 = 2$
- $f_4 = f_3 + f_2 = 2 + 1 = 3$
- $f_5 = f_4 + f_3 = 3 + 2 = 5$

In Chapter 8, we will use the Fibonacci numbers to model population growth of rabbits. This was an application described by Fibonacci himself.

Next, we use strong induction to prove a result about the Fibonacci numbers.

Fibonacci Numbers

Example 4: Show that whenever $n \geq 3$, $f_n > \alpha^{n-2}$, where $\alpha = (1 + \sqrt{5})/2$.

Solution: Let $P(n)$ be the statement $f_n > \alpha^{n-2}$. Use strong induction to show that $P(n)$ is true whenever $n \geq 3$.

- BASIS STEP: $P(3)$ holds since $\alpha < 2 = f_3$
  $P(4)$ holds since $\alpha^2 = (3 + \sqrt{5})/2 < 3 = f_4$.
- INDUCTIVE STEP: Assume that $P(j)$ holds, i.e., $f_j > \alpha^{j-2}$ for all integers j with $3 \leq j \leq k$, where $k \geq 4$. Show that $P(k + 1)$ holds, i.e., $f_{k+1} > \alpha^{k-1}$.
  - Since $\alpha^2 = \alpha + 1$ (because α is a solution of $x^2 - x - 1 = 0$),

    $$\alpha^{k-1} = \alpha^2 \cdot \alpha^{k-3} = (\alpha + 1) \cdot \alpha^{k-3} = \alpha \cdot \alpha^{k-3} + 1 \cdot \alpha^{k-3} = \alpha^{k-2} + \alpha^{k-3}$$

  - By the inductive hypothesis, because $k \geq 4$ we have

    $$f_{k-1} > \alpha^{k-3}, \qquad f_k > \alpha^{k-2}.$$

  - Therefore, it follows that

    $$f_{k+1} = f_k + f_{k-1} > \alpha^{k-2} + \alpha^{k-3} = \alpha^{k-1}.$$

  - Hence, $P(k + 1)$ is true.

Why does this equality hold?

Lamé's Theorem

Gabriel Lamé
(1795-1870)

Lamé's Theorem: Let a and b be positive integers with $a \geq b$. Then the number of divisions used by the Euclidian algorithm to find gcd(a,b) is less than or equal to five times the number of decimal digits in b.

Proof: When we use the Euclidian algorithm to find gcd(a,b) with $a \geq b$,

- n divisions are used to obtain (with $a = r_0, b = r_1$):

$$\begin{aligned}
r_0 &= r_1 q_1 + r_2 && 0 \leq r_2 < r_1,\\
r_1 &= r_2 q_2 + r_3 && 0 \leq r_3 < r_2,\\
&\vdots\\
r_{n-2} &= r_{n-1} q_{n-1} + r_n && 0 \leq r_n < r_{n-1},\\
r_{n-1} &= r_n q_n.
\end{aligned}$$

- Since each quotient $q_1, q_2, \ldots, q_{n-1}$ is at least 1 and $q_n \geq 2$:

$$\begin{aligned}
r_n &\geq 1 = f_2,\\
r_{n-1} &\geq 2\, r_n \geq 2\, f_2 = f_3,\\
r_{n-2} &\geq r_{n-1} + r_n \geq f_3 + f_2 = f_4,\\
&\vdots\\
r_2 &\geq r_3 + r_4 \geq f_{n-1} + f_{n-2} = f_n,\\
b = r_1 &\geq r_2 + r_3 \geq f_n + f_{n-1} = f_{n+1}.
\end{aligned}$$

continued →

Lamé's Theorem

- It follows that if n divisions are used by the Euclidian algorithm to find gcd(a,b) with $a \geq b$, then $b \geq f_{n+1}$.

By Example 4, $f_{n+1} > \alpha^{n-1}$, for $n > 2$, where $\alpha = (1 + \sqrt{5})/2$. Therefore, $b > \alpha^{n-1}$.

- Because $\log_{10} \alpha \approx 0.208 > 1/5$, $\log_{10} b > (n-1) \log_{10} \alpha > (n-1)/5$. Hence,

$$n-1 < 5 \cdot \log_{10} b.$$

- Suppose that b has k decimal digits. Then $b < 10^k$ and $\log_{10} b < k$. It follows that $n - 1 < 5k$ and since k is an integer, $n \leq 5k$.

- As a consequence of Lamé's Theorem, $O(\log b)$ divisions are used by the Euclidian algorithm to find gcd(a,b) whenever $a > b$.
  - By Lamé's Theorem, the number of divisions needed to find gcd(a,b) with $a > b$ is less than or equal to $5(\log_{10} b + 1)$ since the number of decimal digits in b (which equals $\lfloor \log_{10} b \rfloor + 1$) is less than or equal to $\log_{10} b + 1$.

Lamé's Theorem was the first result in computational complexity

Wilson’s Theorem

Theorem:

$$p \text{ is prime}, (p-1)! \equiv -1 \; mod \; p$$

Proof:

Fermat’s Little Theorem

Pierre de Fermat
(1601-1665)

Theorem 3: (*Fermat’s Little Theorem*)
If p is prime and a is an integer not divisible by p, then $a^{p-1} \equiv 1 \pmod{p}$

Furthermore, for every integer a we have $a^p \equiv a \pmod{p}$
(*proof outlined in Exercise 19*)

Fermat’s little theorem is useful in computing the remainders modulo p of large powers of integers.

Example: Find 7^{222} **mod** 11.

By Fermat’s little theorem, we know that $7^{10} \equiv 1 \pmod{11}$, and so $(7^{10})^k \equiv 1 \pmod{11}$, for every positive integer k. Therefore,

$$7^{222} = 7^{22 \cdot 10 + 2} = (7^{10})^{22} 7^2 \equiv (1)^{22} \cdot 49 \equiv 5 \pmod{11}.$$

Hence, 7^{222} **mod** $11 = 5$.

Pseudoprimes

- By Fermat's little theorem $n > 2$ is prime, where

$$2^{n-1} \equiv 1 \pmod{n}.$$

- But if this congruence holds, n may not be prime. Composite integers n such that $2^{n-1} \equiv 1 \pmod{n}$ are called *pseudoprimes* to the base 2.

  Example: The integer 341 is a pseudoprime to the base 2.

  $341 = 11 \cdot 31$

  $2^{340} \equiv 1 \pmod{341}$ (*see in Exercise* 37)

- We can replace 2 by any integer $b \geq 2$.

  Definition: Let b be a positive integer. If n is a composite integer, and $b^{n-1} \equiv 1 \pmod{n}$, then n is called a *pseudoprime to the base b*.

Pseudoprimes

- Given a positive integer *n*, such that $2^{n-1} \equiv 1 \pmod{n}$:
  - If *n* does not satisfy the congruence, it is composite.
  - If *n* does satisfy the congruence, it is either prime or a pseudoprime to the base 2.
- Doing similar tests with additional bases *b*, provides more evidence as to whether *n* is prime.
- Among the positive integers not exceeding a positive real number *x*, compared to primes, there are relatively few pseudoprimes to the base *b*.
  - For example, among the positive integers less than 10^{10} there are $455{,}052{,}512$ primes, but only $14{,}884$ pseudoprimes to the base 2.

Carmichael Numbers

Robert Carmichael
(1879-1967)

- There are composite integers *n* that pass all tests with bases *b* such that gcd(*b,n*) = 1.

  Definition: A composite integer n that satisfies the congruence $b^{n-1} \equiv 1 \pmod{n}$ for all positive integers *b* with gcd(*b*,*n*) = 1 is called a *Carmichael* number.

  Example: The integer 561 is a Carmichael number. To see this:
  - 561 is composite, since 561 = 3 · 11 · 13.
  - If $\gcd(b, 561) = 1$, then $\gcd(b, 3) = 1$, then $\gcd(b, 11) = \gcd(b, 17) = 1$.
  - Using Fermat's Little Theorem: $b^2 \equiv 1 \pmod 3$, $b^{10} \equiv 1 \pmod{11}$, $b^{16} \equiv 1 \pmod{17}$.
  - Then

    $b^{560} = (b^2)^{280} \equiv 1 \pmod 3$,
    $b^{560} = (b^{10})^{56} \equiv 1 \pmod{11}$,
    $b^{560} = (b^{16})^{35} \equiv 1 \pmod{17}$.
  - It follows (*see Exercise* 29) that $b^{560} \equiv 1 \pmod{561}$ for all positive integers *b* with gcd(*b*,561) = 1. Hence, 561 is a Carmichael number.

- Even though there are infinitely many Carmichael numbers, there are other tests (described in the exercises) that form the basis for efficient probabilistic primality testing. (*see Chapter* 7)

Additional Reads

Reads:

- Chap. 4 and 5, Discrete Math. and Its applications, K.H. Rosen
- Chap. 4.8, Handbook of Discrete and Combinatorial Math., K.H. Rosen