CS 2305: Discrete Mathematics for Computing I

Lecture 22

- KP Bhat

Number Theory ~~and Cryptography~~

Chapter 4

Background Information

Number theory is the part of mathematics devoted to the study of the integers and their properties.

Number theory has many applications in computer science including cryptography, pseudorandom number generation, error detection and error correction codes etc.

Many computer applications of number theory are based on the notion of divisibility and primality of integers.

Divisibility and Modular Arithmetic

Section 4.1

Division [1]

Definition: If a and b are integers with $a \neq 0$, then a *divides* b if there exists an integer c such that $b = ac$.

- When a divides b we say that a is a *factor* or *divisor* of b and that b is a multiple of a.
- The notation $a \mid b$ denotes that a divides b.
- If $a \mid b$, then b/a is an integer.
- If a does not divide b, we write $a \nmid b$.

For example: $3 \nmid 7$ and $3 \mid 12$.

Division $_2$

- $a \mid b$ is a Boolean function
  - returns True of False
- $a \mid b$ is true if

  $b \div a = q_{\text{quotient}}$ and $0_{\text{remainder}}$ $\therefore b = aq$
- $a \mid b$ is false (i.e. $a \nmid b$) if

  $b \div a = q_{\text{quotient}}$ and $r(\neq 0)_{\text{remainder}}$ $\therefore b = aq + r$

Properties of Divisibility

Theorem 1: Let a, b, and c be integers, where $a \neq 0$.

i. If $a \mid b$ and $a \mid c$, then $a \mid (b + c)$;

ii. If $a \mid b$, then $a \mid bc$ for all integers c;

iii. If $a \mid b$ and $b \mid c$, then $a \mid c$.

Proof: (i) Suppose $a \mid b$ and $a \mid c$, then it follows that there are integers s and t with $b = as$ and $c = at$. Hence,

$b + c = as + at = a(s + t)$. Hence, $a \mid (b + c)$

(Exercises 3 and 4 ask for proofs of parts (ii) and (iii).)

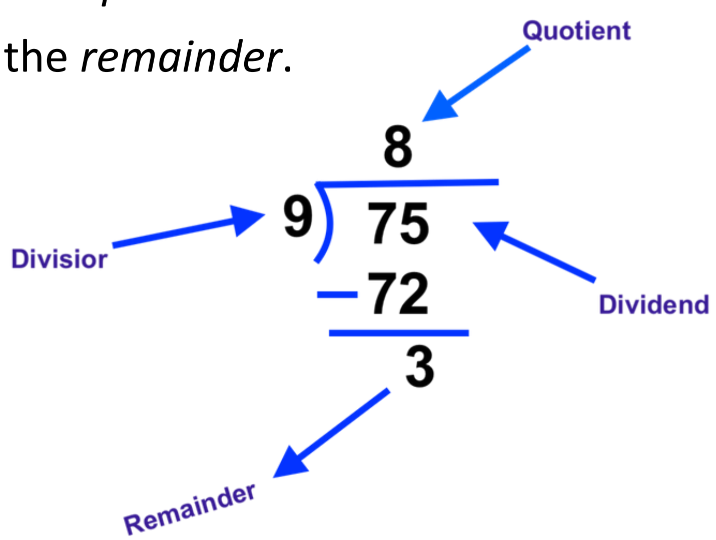
Corollary: If a, b, and c be integers, where $a \neq 0$, such that $a \mid b$ and $a \mid c$, then $a \mid mb + nc$ whenever m and n are integers.

Division Algorithm[1]

When an integer is divided by a positive integer, there is a quotient and a remainder. This is traditionally called the "Division Algorithm," but is really a theorem.

Division Algorithm: If a is an integer and <u>d a positive integer</u>, then there are unique integers q and r, with $0 \le r < d$, such that $a = dq + r$ (*proved in Section* 5.2).

- d is called the *divisor*.
- a is called the *dividend*.
- q is called the *quotient*.
- r is called the *remainder*.

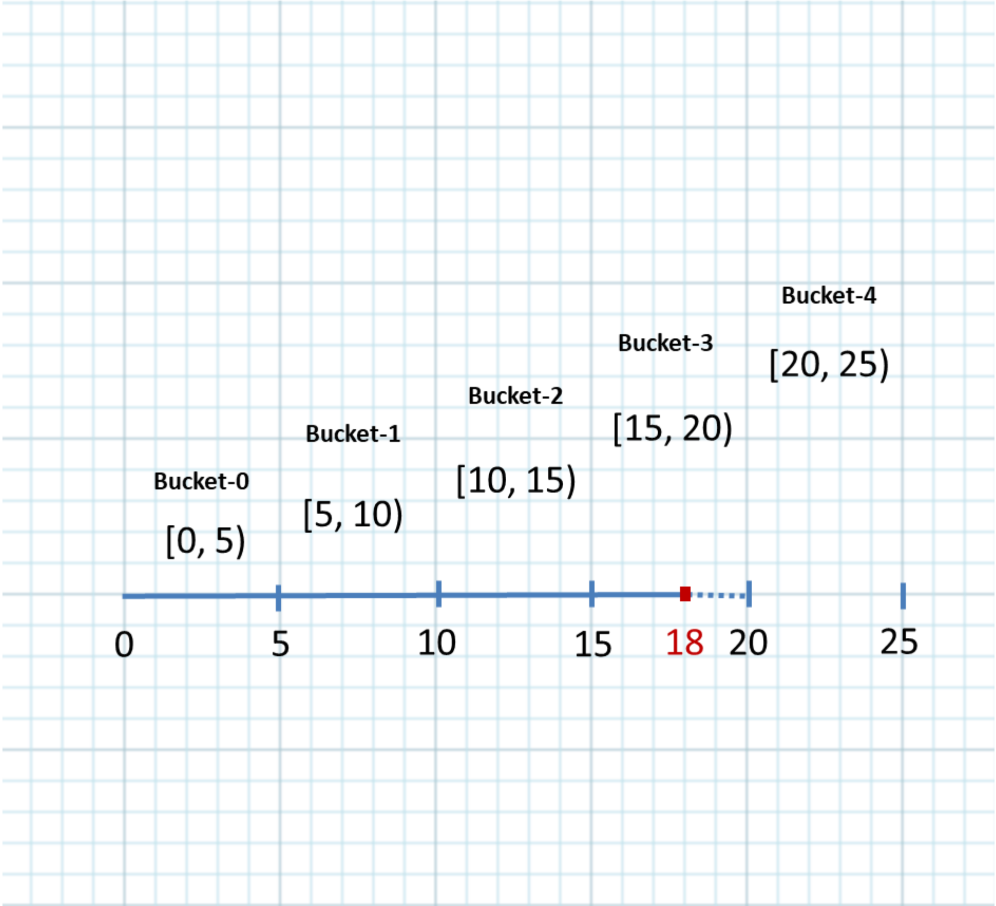


Definitions of Functions **div** and **mod**

$q = a$ **div** d

$r = a$ **mod** d

+ve and –ve Remainders



- 18/5 = 3 remainder 3
  - 3 more than the current bucket boundary
- 18/5 = 4 remainder -2
  - 2 less than the next bucket boundary
- From this perspective division will either yield a remainder of 0 or it will yield a +ve and a –ve remainder
- We will only consider the +ve remainder

Division Algorithm[2]

Examples:

– What are the quotient and remainder when 101 is divided by 11?
  - Solution:
    – Quotient = 101 div 11 = 9
    – Remainder = 101 mod 11 = 2

– What are the quotient and remainder when −11 is divided by 3?
  - Solution
    – Quotient = -11 div 3 = -4
    – Remainder = -11 mod 3 = 1

Remember we only consider the +ve remainder

Elementary School Approach
Quotient: -3
Remainder: -2

$q = a$ **div** d
$r = a$ **mod** d

Procedure for +ve Remainder

- Given dividend a and divisor d
  - Step 1: First find the quotient
    - q=$\lfloor \frac{a}{d} \rfloor$
      - Remember that $\lfloor$-x$\rfloor$ = -$\lceil$x$\rceil$
  - Step 2: Find the +ve remainder
    - r = a – d*q
- For -11 divided by 3
  - Here a = -11 and d = 3
    - q=$\lfloor \frac{-11}{3} \rfloor$ = - $\lceil \frac{11}{3} \rceil$ = -4
    - r = -11 – 3(-4) = -11 + 12 = 1

Notes:
We only consider +ve divisors

If the dividend is +ve, you use your elementary school math approach

If the dividend is –ve, the elementary school math approach will give a –ve remainder, which we don't want. In that case you use this approach

Congruence Relation[1]

- Two integers a and b are said to be *congruent modulo m* [represented as *$a \equiv b$ (mod m)*] if a **mod** m = b **mod** m
  - Same remainder when they are divided by the positive integer m
  - $18 \equiv 53 \pmod 7$
    - "18 is congruent to 53 modulo 7"

Let $a \equiv b \pmod m$

Then a and b can be represented as:

$a = q_1 m + r$; $b = q_2 m + r$

Clearly $(a - b) = (q_1 m + r) - (q_2 m + r) = (q_1 - q_2)m$

$\therefore m \mid (a-b)$

This forms the basis for the formal definition of congruence

Congruence Relation₂

Definition: If a and b are integers and m is a positive integer, then a is *congruent* to b *modulo* m if m divides $a - b$.

- The notation $a \equiv b \pmod{m}$ says that a is congruent to b modulo m.
- We say that $a \equiv b \pmod{m}$ is a *congruence* and that m is its *modulus.*
- Two integers are congruent mod m if and only if they have the same remainder when divided by m.
- If a is not congruent to b modulo m, we write $a \not\equiv b \pmod{m}$

Example: Determine whether 17 is congruent to 5 modulo 6 and whether 24 and 14 are congruent modulo 6.

Solution:

- $17 \equiv 5 \pmod{6}$ because 6 divides $17 - 5 = 12$.
- $24 \not\equiv 14 \pmod{6}$ since $24 - 14 = 10$ is not divisible by 6.

More on Congruences

Theorem 4: Let m be a positive integer. The integers a and b are congruent modulo m if and only if there is an integer k such that $a = b + km$.

Proof:

- If $a \equiv b \pmod{m}$, then (by the definition of congruence) $m \mid a - b$. Hence, there is an integer k such that $a - b = km$ and equivalently $a = b + km$.
- Conversely, if there is an integer k such that $a = b + km$, then $km = a - b$. Hence, $m \mid a - b$ and $a \equiv b \pmod{m}$.

Note on proofs

- You can derive most of the proofs in this section using elementary school concepts
  - Only beware of the –ve remainder problem with the –ve dividend
- In many of the proofs you will start by considering the dividend (say a), the divisor (say m), the quotient (say q) and the remainder (say r)
- Keep in mind the following:
  - $a = q * m + r$
  - $m \mid (a-r)$
- Given m | x then
  - $x = k * m$
- Given a≡b (mod m) then
  - a mod m = b mod m (i.e same remainder on division)
  - m | (a – b)
  - a = b + km

Congruence Class

- The set of all integers congruent to *a mod m* is called the **congruence class** of a modulo m
  - For example the congruence class for 3 mod 4 is $\{\ldots,-5,-1, 3, 7, 11,\ldots\}$

Congruences of Sums and Products

Theorem 5: Let m be a positive integer. If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$ and $ac \equiv bd \pmod{m}$

Proof:

- Because $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, by Theorem 4 there are integers s and t with $b = a + sm$ and $d = c + tm$.
- Therefore,
  - $b + d = (a + sm) + (c + tm) = (a + c) + m(s + t)$ and
  - $b\,d = (a + sm)\,(c + tm) = ac + m(at + cs + stm)$.
- Hence, $a + c \equiv b + d \pmod{m}$ and $ac \equiv bd \pmod{m}$.

Theorem 4

$a \equiv b \pmod{m} \leftrightarrow a = b + km$

Example: Because $7 \equiv 2 \pmod{5}$ and $11 \equiv 1 \pmod{5}$, it follows from Theorem 5 that

$$18 = 7 + 11 \equiv 2 + 1 = 3 \pmod{5}$$

$$77 = 7 \cdot 11 \equiv 2 \cdot 1 = 2 \pmod{5}$$

Algebraic Manipulation of Congruences

Multiplying both sides of a valid congruence by an integer preserves validity.

- If $a \equiv b \pmod{m}$ holds then $c \cdot a \equiv c \cdot b \pmod{m}$, where c is any integer, holds by Theorem 5 with $d = c$.

Adding an integer to both sides of a valid congruence preserves validity.

- If $a \equiv b \pmod{m}$ holds then $c + a \equiv c + b \pmod{m}$, where c is any integer, holds by Theorem 5 with $d = c$.

Dividing a congruence by an integer does not always produce a valid congruence.

- Theorem 7 Section 4.3 provides the conditions when division is ok.

Example: The congruence $14 \equiv 8 \pmod{6}$ holds. But dividing both sides by 2 does not produce a valid congruence since $14/2 = 7$ and $8/2 = 4$, but $7 \not\equiv 4 \pmod{6}$.

Computing the **mod** *m* Function of Products and Sums[1]

Corollary: Let m be a positive integer and let a and b be integers. Then
$(a + b)$ (**mod** m) = ((a **mod** m) + (b **mod** m)) **mod** m
and
ab **mod** m = ((a **mod** m) (b **mod** m)) **mod** m

Proof:

Step 1: Prove that a ≡ (a **mod** m) (**mod** m)

By definition we know that a ≡ b (**mod** m) if m | (a – b)

Let a **mod** m = r

Then a = mq + r and r = a – mq, for some quotient q

a – r = a – (a – mq) = mq

Clearly m | (a – r) so a ≡ r (**mod** m)

∴ a ≡ (a **mod** m) (**mod** m)

Similarly b ≡ (b **mod** m) (**mod** m)

Computing the **mod** *m* Function of Products and Sums[2]

Proof (Cont'd):

Step 2: Use the Congruence Addition and Multiplication Theorem (Theorem 5)

If a ≡ b (**mod** m) and c ≡ d (**mod** m), then
i. a + c ≡ b + d (**mod** m)
ii. ac ≡ bd (**mod** m).

$$a + b \equiv ((a \ \mathbf{mod}\ m) + (b \ \mathbf{mod}\ m))\ (\mathbf{mod}\ m)$$

$$ab \equiv ((a \ \mathbf{mod}\ m)(b \ \mathbf{mod}\ m))\ (\mathbf{mod}\ m)$$

Step 3: Apply Theorem 3

a ≡ b (**mod** m) if and only if
a **mod** m = b **mod** m

$$(a + b)(\mathbf{mod}\ m) = ((a \ \mathbf{mod}\ m) + (b \ \mathbf{mod}\ m))\ (\mathbf{mod}\ m)$$

$$ab \ \mathbf{mod}\ m = ((a \ \mathbf{mod}\ m)(b \ \mathbf{mod}\ m))\ (\mathbf{mod}\ m)$$

Example of mod computation

Example: Find the value of $(19^3$ **mod** $31)^4$ **mod** 23.

Solution: To compute $(19^3$ **mod** $31)^4$ **mod** 23, we will first evaluate 19^3 **mod** 31.

Because $19^3 = 6859$ and $6859 = 221 \cdot 31 + 8$, we have 19^3 **mod** $31 = 6859$ **mod** $31 = 8$.

So, $(19^3$ **mod** $31)^4$ **mod** $23 = 8^4$ **mod** 23.

Next, note that $8^4 = 4096$. Because $4096 = 178 \cdot 23 + 2$,we have 4096 **mod** $23 = 2$. Hence,

$(19^3$ **mod** $31)^4$ **mod** $23 = 2$.

Arithmetic Modulo m

Definitions: Let $\mathbf{Z}_m$ be the set of nonnegative integers less than m: $\{0,1, \ldots., m-1\}$

- The operation $+_m$ is defined as $a +_m b = (a + b)$ **mod** m. This is *addition modulo m*.
- The operation $\cdot_m$ is defined as $a \cdot_m b = (a \cdot b)$ **mod** m. This is *multiplication modulo m*.
- Using these operations is said to be doing *arithmetic modulo m*.

Example: Find $7 +_{11} 9$ and $7 \cdot_{11} 9$.

Solution: Using the definitions above:

- $7 +_{11} 9 = (7 + 9)$ **mod** $11 = 16$ **mod** $11 = 5$
- $7 \cdot_{11} 9 = (7 \cdot 9)$ **mod** $11 = 63$ **mod** $11 = 8$

Applications of Congruences

For this section only concepts, no computations, on the exam

Section 4.5

Background Information

- Congruences have many applications in discrete mathematics, computer science, and many other disciplines
- We will discuss three applications
  - the use of congruences to assign memory locations to data records
  - the generation of pseudorandom numbers
  - check digits
- Congruences also play an extremely important role in cryptography

Hashing Functions[1]

Definition: A *hashing function* h assigns memory location $h(k)$ to a record that has k as its key.

- A common hashing function is $h(k) = k \bmod m$, where m is the number of memory locations.
- Because this hashing function is onto, all memory locations are possible.

- The hashing function is not one-to-one as there are many more possible keys than memory locations. When more than one record is assigned to the same location, we say a collision occurs.
- When a collision takes place, a rehash strategy is executed
  - For example, collision can be resolved by assigning the record to the first free location.
  - In this case we can use a *linear probing function*: $h(k,i) = (h(k) + i)$ **mod** m, where i runs from 0 to $m - 1$.
  - There are many other implementations for the rehash strategy (quadratic probing, double hashing etc.)
    - These will be covered in advanced CS courses

Hashing Functions[2]

Example: Let $h(k) = k \bmod 111$. This hashing function assigns the records of customers with social security numbers as keys to memory locations in the following manner:

$h(064212848) = 064212848 \bmod 111 = 14$

$h(037149212) = 037149212 \bmod 111 = 65$

$h(107405723) = 107405723 \bmod 111 = 14$, but since location 14 is already occupied, the record is assigned to the next available position, which is 15.

0	
1	
2	
…	
14	064212848
15	107405723
…	
65	037149212
…	
110	

Pseudorandom Numbers [1]

Randomly chosen numbers are needed for many purposes, including computer simulations.

Pseudorandom numbers are not truly random since they are generated by systematic methods but they strive for some desirable properties of random numbers like uniformity and independence

The *linear congruential method* is one commonly used procedure for generating pseudorandom numbers.

Four integers are needed:

i. the *modulus m*,
ii. the *multiplier a*,
iii. the *increment c*,
iv. *the seed* x_0,

with $2 \leq a < m$, $0 \leq c < m$, $0 \leq x_0 < m$.

We generate a sequence of pseudorandom numbers $\{x_n\}$, with $0 \leq x_n < m$ for all n, by successively using the recursively defined function

$$x_{n+1} = (ax_n + c) \textbf{ mod } m.$$

Pseudorandom Numbers [2]

Example: Find the sequence of pseudorandom numbers generated by the linear congruential method with modulus $m = 9$, multiplier $a = 7$, increment $c = 4$, and seed $x_0 = 3$.

Solution: Compute the terms of the sequence by successively using the congruence

$$x_{n+1} = (7x_n + 4) \textbf{ mod } 9, \text{ with } x_0 = 3.$$

$$
\begin{aligned}
x_1 &= 7x_0 + 4 \textbf{ mod } 9 = 7\cdot 3 + 4 \textbf{ mod } 9 = 25 \textbf{ mod } 9 = 7,\\
x_2 &= 7x_1 + 4 \textbf{ mod } 9 = 7\cdot 7 + 4 \textbf{ mod } 9 = 53 \textbf{ mod } 9 = 8,\\
x_3 &= 7x_2 + 4 \textbf{ mod } 9 = 7\cdot 8 + 4 \textbf{ mod } 9 = 60 \textbf{ mod } 9 = 6,\\
x_4 &= 7x_3 + 4 \textbf{ mod } 9 = 7\cdot 6 + 4 \textbf{ mod } 9 = 46 \textbf{ mod } 9 = 1,\\
x_5 &= 7x_4 + 4 \textbf{ mod } 9 = 7\cdot 1 + 4 \textbf{ mod } 9 = 11 \textbf{ mod } 9 = 2,\\
x_6 &= 7x_5 + 4 \textbf{ mod } 9 = 7\cdot 2 + 4 \textbf{ mod } 9 = 18 \textbf{ mod } 9 = 0,\\
x_7 &= 7x_6 + 4 \textbf{ mod } 9 = 7\cdot 0 + 4 \textbf{ mod } 9 = 4 \textbf{ mod } 9 = 4,\\
x_8 &= 7x_7 + 4 \textbf{ mod } 9 = 7\cdot 4 + 4 \textbf{ mod } 9 = 32 \textbf{ mod } 9 = 5,\\
x_9 &= 7x_8 + 4 \textbf{ mod } 9 = 7\cdot 5 + 4 \textbf{ mod } 9 = 39 \textbf{ mod } 9 = 3.
\end{aligned}
$$

The sequence generated is 3,7,8,6,1,2,0,4,5,3,7,8,6,1,2,0,4,5,3,…

It repeats after generating 9 terms.

Commonly, computers use a linear congruential generator with increment $c = 0$. This is called a *pure multiplicative generator*. Such a generator with modulus $2^{31} - 1$ and multiplier $7^5 = 16{,}807$ generates $2^{31} - 2$ numbers before repeating.

Pseudorandom Numbers 3

- If psuedo-random numbers between 0 and 1 are needed, then the generated numbers are divided by the modulus, x_n/m
- Linear congruential generators provide a very efficient way to generate pseudo-random numbers which are suitable for many applications
- Unfortunately long pseudo-random number sequences do not share some important statistical properties that true random numbers have. Because of this, it is not advisable to use them for some tasks, such as large simulations