Discrete Structures
CSC160

Instructor: Prakash Neupane
Godawari College
Itahari

Integers and Matrices

- Chapter outline:
  - Integers and Division
  - Primes and Greatest Common Divisor
  - Extended Euclidean Algorithm
  - Integers and Algorithms
  - Applications of Number Theory
    - Linear Congruencies
    - Chinese Remainder Theorem
    - Computer Arithmetic with Large Integers
  - Matrices: Zero-One Matrices, Boolean Matrix Operations

Integers and Matrices

- Integers and Division :
  - When one integer is divided by a second nonzero integer, the quotient may or may not be an integer.
  - For example, 12/3 = 4 is an integer, whereas 11/4 = 2.75 is not.

  - ***If a and b are integers with a = 0, we say that a divides b if there is 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 | b denotes that a divides b. We write a | b when a does not divide b.***

Integers and Matrices

- Integers and Division : THE DIVISION ALGORITHM
  - *Let, **a** be an integer and **d** a positive integer. Then there are unique integers **q** and **r**, with $\mathbf{0 \leq r < d}$, such that $\mathbf{a = dq + r}$.*
  - *Where, In the equality given in the division algorithm,*
    - *d is called the divisor,*
    - *a is called the dividend,*
    - *q is called the quotient, and*
    - *r is called the remainder.*
  - *This notation is used to express the quotient and remainder:*
    - *q = a div d,*
    - *r = a mod d*

- Integers and Division : THE DIVISION ALGORITHM
  - *What are the quotient and remainder when 101 is divided by 11?*
  - *Solution: We have, such that* $\mathbf{a = dq + r}$*.*
    - $101 = 11 \cdot 9 + 2.$
  - *Hence,*
    - *the quotient when 101 is divided by 11 is 9 = 101 div 11, and*
    - *the remainder is 2 = 101 mod 11.*

Integers and Matrices

- Modular Arithmetic:
  - In some situations we care only about the remainder of an integer when it is divided by some specified positive integer.
  - For instance, when we ask what time it will be (on a 24-hour clock) 50 hours from now, we care only about the remainder when 50 plus the current hour is divided by 24.

Integers and Matrices

- Modular Arithmetic:
  - If a and b are integers and m is a positive integer, then a is congruent to b modulo m if m divides $a - b$.
  - We use the notation $a \equiv b \pmod{m}$ to indicate 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 (plural moduli).
  - If a and b are not congruent modulo m, we write $a \equiv b \pmod{m}$.

Integers and Matrices

- Modular Arithmetic:
  - Determine whether 17 is congruent to 5 modulo 6 and whether 24 and 14 are congruent modulo 6.
  - Solution:
    - Because 6 divides $17 - 5 = 12$, we see that $17 \equiv 5 \pmod 6$.
    - However, because $24 - 14 = 10$ is not divisible by 6, we see that $24 \not\equiv 14 \pmod 6$.

- Primes and Greatest Common Divisor
  - An integer p greater than 1 is called prime if the only positive factors of p are 1 and p.
  - A positive integer that is greater than 1 and is not prime is called composite.
  - Example:
  - The integer 7 is prime because its only positive factors are 1 and 7,
  - whereas the integer 9 is composite because it is divisible by 3.

Integers and Matrices

- Primes and Greatest Common Divisor
  - Algorithm for Finding Prime Numbers:
  - To determine if an integer n is prime:
    - If n is less than 2, it is not prime.
    - Check for divisors of n from 2 up to $\sqrt{n}$ (square root of n).
    - If n is divisible by any number in this range, it is not prime.
    - Otherwise, n is prime

- Primes and Greatest Common Divisor
  - Algorithm for Finding Prime Numbers:
  - Example 1: Prime Numbers
    - Let's check if $n = 17$ is prime:
    - $\sqrt{17} \approx 4.123$, so we check for divisors from 2 to 4.
    - 17 is not divisible by any number between 2 and 4, so it is prime.

- Primes and Greatest Common Divisor
  - **THE FUNDAMENTAL THEOREM OF ARITHMETIC**
  - Every integer greater than 1 can be written uniquely as a prime or as the product of two or more primes where the prime factors are written in order of nondecreasing size.
  - Example:
    - The prime factorizations of 100 is:
    - $100 = 2 \cdot 2 \cdot 5 \cdot 5 = 2^2\, 5^2$

Integers and Matrices

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - The largest integer that divides both of two integers is called the greatest common divisor of these integers.
  - *Let a and b be integers, not both zero. The largest integer d such that d | a and d | b is called the greatest common divisor of a and b.*
  - *The greatest common divisor of a and b is denoted by gcd(a, b).*

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - *What is the greatest common divisor of 24 and 36?*

    Solution:

    The positive common divisors of 24 and 36 are 1, 2, 3, 4, 6, and 12.

    Hence, gcd(24, 36) = 12.

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - *What is the greatest common divisor of 17 and 22?*
  - *Solution:*
  - *The integers 17 and 22 have no positive common divisors other than 1,*
  - *so that gcd(17, 22) = 1.*
  - *The integers a and b are relatively prime if their greatest common divisor is 1.*

Integers and Matrices

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - To find the GCD of two integers a and b:
  - *If b is zero,*
    - *return a as the GCD.*
  - *Otherwise,*
    - *recursively call the GCD function with arguments b and a % b, where % denotes the modulo operator.*
  - *Repeat until b becomes zero.*

Integers and Matrices

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - Example 3: Greatest Common Divisor (GCD)
    - a = 24, b = 36
    - **GCD(24, 36) = GCD(36, 24) = GCD(24, 12) = GCD(12, 0) = 12**

Integers and Matrices

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - Example: gcd(48,18); a= b*q+r
    -

A=48, b=18	
48 = 18*2+12	Largest number on the left
18 = 12*1 +6	sift left
12= 6*2+0	

    - Gcd(48,18) is 6; it is last non zero remainder.

Integers and Matrices

- Greatest Common Divisor (GCD)/ Euclidean Algorithm
  - Python Implementation

```python
def gcd(b, a):
    if b == 0:
        return a
    return gcd(a % b, b)

a = 24
b = 36
print("gcd(", a, ",", b, ") = ", gcd(b, a))
```

- Extended Euclidean Algorithm
  - Extended Euclidean algorithm also finds integer coefficients x and y such that:
    - $ax + by = \gcd(a, b)$
  - Example:
    - Input: $a = 30$, $b = 20$
    - Output: $\gcd = 10$, $x = 1$, $y = -1$
    - (Note that $30*1 + 20*(-1) = 10$)

Integers and Matrices

- Extended Euclidean Algorithm : Python

```
def gcdExtended(a,b):
  if a==0:
    d=a
    x=1
    y=0
    return d,x,y
  x2=1
  x1=0
  y1=1
  y2=0
while(b>0):
    q=(a//b)
    r=a-q*b
    x=x2-q*x1
    y=y2-q*y1
    a=b
    b=r
    x2=x1
    x1=x
    y2=y1
    y1=y
d=a
x=x2
y=y2
return d,x,y
```

```
a=21
b=15
g, x,y =gcdExtended(a,b)
print("gcd(", a, ",", b, ") = ", g, x, y)
```

Output:

gcd(21 , 15) = 3 -2 3

- **Extended Euclidean Algorithm : Example**
- Let's find the coefficients (x and y) for a = 21 and b = 15 such that 21x + 15y = GCD(21, 15):

q	a	b	r	x1	x2	x	y1	y2	y

Integers and Matrices

- Integers and Algorithms: Representation of Integer
  - In everyday life we use decimal notation to express integers.
  - For example, 965 is used to denote $9 \cdot 10^2 + 6 \cdot 10 + 5$.
  - However, it is often convenient to use bases other than 10.
  - In particular, computers usually use binary notation (with 2 as the base) when carrying out arithmetic, and octal (base 8) or hexadecimal (base 16) notation when expressing characters, such as letters or digits.

Integers and Matrices

- Integers and Algorithms: Representation of Integer
  - Formally,
  - Let b be an integer greater than 1.
  - Then if n is a positive integer, it can be expressed uniquely in the form
    - $n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b + a0$,
  - where k is a nonnegative integer, a0 , a1 , . . . , ak are nonnegative integers less than b, and $a_k = 0$.

Integers and Matrices

- Integers and Algorithms: Representation of Integer
  - **What is the decimal expansion of the integer that has $(10101\ 1111)_2$ as its binary expansion?**
  - Solution:
  - We have
  - (1 0101 1111)2
  - $= 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$
  - $= (351)_{10}$.

Integers and Matrices

- Integers and Algorithms: Representation of Integer
  - **What is the decimal expansion of the number with octal expansion $(7016)_8$?**
  - **What is the decimal expansion of the number with hexadecimal expansion $(2AE0B)_{16}$?**

- Integers and Algorithms: BASE CONVERSION
  - Base b expansion of an integer n.
  - First, divide n by b to obtain a quotient and remainder, that is,
  - $n = bq_0 + a_0$, such as $0 \le a_0 < b$.
  - The remainder, a_0 , is the rightmost digit in the base b expansion of n. Next, divide q_0 by b to obtain
  - $q_0 = bq_1 + a_1$, Such as $0 \le a_1 < b$.
  - We see that a_1 is the second digit from the right in the base b expansion of n. Continue this process, successively dividing the quotients by b, obtaining additional base b digits as the remainders. This process terminates when we obtain a quotient equal to zero. It produces the
  - base b digits of n from the right to the left.

- Integers and Algorithms: BASE CONVERSION
  - Find the octal expansion of $(12345)_{10}$.
  - Solution: First, divide 12345 by 8 to obtain
  - $12345 = 8 \cdot 1543 + 1$.
  - Successively dividing quotients by 8 gives
  - $1543 = 8 \cdot 192 + 7$,
  - $192 = 8 \cdot 24 + 0$,
  - $24 = 8 \cdot 3 + 0$,
  - $3 = 8 \cdot 0 + 3$.
  - The successive remainders that we have found, 1, 7, 0, 0, and 3, are the digits from the right to the left of 12345 in base 8. Hence,
  - $(12345)_{10} = (30071)_8$.

- Integers and Algorithms: BASE CONVERSION
  - Find the hexadecimal expansion of $(177130)_{10}$.
  - Find the binary expansion of $(241)_{10}$.

Integers and Matrices

- Integers and Algorithms: Algorithms for Integer Operations
  - The algorithms for performing operations with integers using their binary expansions are extremely important in computer arithmetic.
  - Suppose that the binary expansions of a and b are
    - $a = (a_{n-1}\, a_{n-2}\, . . .\, a_1\, a_0\,)_2$, and
    - $b = (b_{n-1}\, b_{n-2}\, . . .\, b_1\, b_0\,)_2$,
  - so that a and b each have n bits (putting bits equal to 0 at the beginning of one of these expansions if necessary).

Integers and Matrices

- Integers and Algorithms: ADDITION ALGORITHM
  - To add a and b, first add their rightmost bits. This gives,
  - $a_0 + b_0 = c_0 \cdot 2 + s_0$,
  - where s_0 is the rightmost bit in the binary expansion of a + b and c_0 is the carry, which is either 0 or 1.
  - Then add the next pair of bits and the carry: $a_1 + b_1 + c_0 = c_1 \cdot 2 + s_1$
  - where s_1 is the next bit (from the right) in the binary expansion of a + b, and c_1 is the carry.
  - Continue this process, adding the corresponding bits in the two binary expansions and the carry, to determine the next bit from the right in the binary expansion of a + b.

Integers and Matrices

- Integers and Algorithms: ADDITION ALGORITHM
  - Add $a = (1110)_2$ and $b = (1011)_2$.
  - Solution: Following the procedure specified in the algorithm, first note that
  - $a_0 + b_0 = 0 + 1 = 0 \cdot 2 + 1$,
  - so that $c_0 = 0$ and $s_0 = 1$. Then, because
  - $a_1 + b_1 + c_0 = 1 + 1 + 0 = 1 \cdot 2 + 0$,
  - it follows that c1 = 1 and $s_1 = 0$. Continuing,
  - $a_2 + b_2 + c_1 = 1 + 0 + 1 = 1 \cdot 2 + 0$,
  - so that c2 = 1 and $s_2 = 0$. Finally, because
  - $a_3 + b_3 + c_2 = 1 + 1 + 1 = 1 \cdot 2 + 1$,
  - follows that $c_3 = 1$ and $s_3 = 1$. This means that $s_4 = c_3 = 1$. Therefore, $s = a + b = (1\ 1001)_2$.

Integers and Matrices

- Integers and Algorithms: Product ALGORITHM
  - Find the product of $a = (110)_2$ and $b = (101)_2$.
  - Solution: First note that
  - $ab_0 \cdot 2^0 = (110)_2 \cdot 1 \cdot 2^0 = (110)_2$,
  - $ab_1 \cdot 2^1 = (110)_2 \cdot 0 \cdot 2^1 = (0000)_2$,
  - and
  - $ab_2 \cdot 2^2 = (110)_2 \cdot 1 \cdot 2^2 = (11000)_2$.
  - To find the product, add $(110)_2$, $(0000)_2$, and $(11000)_2$. Carrying out these additions (using Addition Algorithm, including initial zero bits when necessary) shows that

    ab = (1 1110)2 .

- Chapter outline: Chinese Remainder Theorem
  - Let m_1 , m_2 , . . . , m_n be pairwise relatively
  - prime positive integers greater than one and a1 , a2 , . . . , an arbitrary integers. Then the system
  - $x \equiv a_1 \pmod{m_1}$,
  - $x \equiv a_2 \pmod{m_2}$,
  - .
  - .
  - $x \equiv a_n \pmod{m_n}$
  - has a unique solution modulo $m = m_1 m_2 \cdots m_n$.
  - (That is, there is a solution x with $0 \leq x < m$, and all other solutions are congruent modulo m to this solution.)

Integers and Matrices

- Chapter outline: 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; and when divided by 7, the remainder is 2.
  - What will be the number of things?
  - This puzzle can be translated into the following question: What are the solutions of the
  - systems of congruences
    - $x \equiv 2 \pmod{3}$,
    - $x \equiv 3 \pmod{5}$,
    - $x \equiv 2 \pmod{7}$?

Integers and Matrices

- Chapter outline: Chinese Remainder Theorem
  - To solve the system of congruences in above example, first
  - let $m = 3 \cdot 5 \cdot 7 = 105$,
  - $M1 = m/3 = 35$,
  - $M2 = m/5 = 21$, and
  - $M3 = m/7 = 15$.
  - We see that 2 is an inverse of $M1 = 35$ modulo 3, because $35 \cdot 2 \equiv 2 \cdot 2 \equiv 1 \pmod 3$;
  - 1 is an inverse of $M2 = 21$ modulo 5, because $21 \equiv 1 \pmod 5$; and
  - 1 is an inverse of $M3 = 15 \pmod 7$, because $15 \equiv 1 \pmod 7$.
  - The solutions to this system are those x such that
  - $x \equiv a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 = 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1$
  - $= 233 \equiv 23 \pmod{105}$.

- Chapter outline: Chinese Remainder Theorem
  - The solutions to this system are those x such that
  - $x \equiv a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 = 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1$
  - $= 233 \equiv 23 \pmod{105}$.
  - It follows that 23 is the smallest positive integer that is a simultaneous solution.
  - We conclude that 23 is the smallest positive integer that leaves a remainder of 2 when divided by 3, a remainder of 3 when divided by 5, and a remainder of 2 when divided by 7.

Integers and Matrices

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations

A ***matrix*** is a rectangular array of objects (usually numbers).

An $m \times n$ ("m by n") matrix has exactly m horizontal rows, and n vertical columns.

$$\begin{bmatrix} 2 & 3 \\ 5 & -1 \\ 7 & 0 \end{bmatrix}$$

A 3×2 matrix

Plural of matrix = *matrices* (say MAY-trih-sees)

An $n \times n$ matrix is called a *square* matrix

Integers and Matrices

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations
- Tons of applications, including:
  - Solving systems of linear equations
  - Computer Graphics, Image Processing
  - Games
  - Models within many areas of
  - Computational Science & Engineering
  - Quantum Mechanics, Quantum Computing
  - Many, many more…

Integers and Matrices

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations
  - Useful for representing other structures.
    - *E.g.*, relations, directed graphs (later on)
  - All elements of a *zero-one* matrix are either 0 or 1.
    - E.g., representing **False** & **True** respectively.
  - The ***join*** of **A**, **B** (both $m \times n$ zero-one matrices):
    - $\mathbf{A} \vee \mathbf{B} = [a_{ij} \vee b_{ij}]$
  - The ***meet*** of **A**, **B**:
    - $\mathbf{A} \wedge \mathbf{B} = [a_{ij} \wedge b_{ij}] = [a_{ij}\, b_{ij}]$

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ and } \boldsymbol{B} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

$$\boldsymbol{A} \vee \boldsymbol{B} = \begin{bmatrix} 1 \vee 0 & 0 \vee 1 & 1 \vee 0 \\ 0 \vee 1 & 1 \vee 1 & 0 \vee 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$\boldsymbol{A} \wedge \boldsymbol{B} = \begin{bmatrix} 1 \wedge 0 & 0 \wedge 1 & 1 \wedge 0 \\ 0 \wedge 1 & 1 \wedge 1 & 0 \wedge 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations
  - Let **A** = $[a_{ij}]$ be an $m \times k$ zero-one matrix and **B** = $[b_{ij}]$ be a $k \times n$ zero-one matrix,
  - The ***boolean product*** of **A** and **B** is like normal matrix multiplication, but using $\vee$ instead of +, and $\wedge$ instead of $\times$ in the row-column "vector dot product":

$$\mathbf{A} \odot \mathbf{B} = \mathbf{C} = [c_{ij}] = \left[\bigvee_{\ell=1}^{k} a_{i\ell} \wedge b_{\ell j} \right]$$

- Find the Boolean product of **A** and **B**, where

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations

$$\mathbf{A} \odot \mathbf{B} = \mathbf{C} = [c_{ij}] = \left[\bigvee_{\ell=1}^{k} a_{i\ell} \wedge b_{\ell j} \right]$$

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 1) \vee (0 \wedge 1) & (1 \wedge 0) \vee (0 \wedge 1) \\ (0 \wedge 1) \vee (1 \wedge 0) & (0 \wedge 1) \vee (1 \wedge 1) & (0 \wedge 0) \vee (1 \wedge 1) \\ (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 1) \vee (0 \wedge 1) & (1 \wedge 0) \vee (0 \wedge 1) \end{bmatrix}$$

Integers and Matrices

- Chapter outline: Zero-One Matrices, Boolean Matrix Operations

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} 1 \vee 0 & 1 \vee 0 & 0 \vee 0 \\ 0 \vee 0 & 0 \vee 1 & 0 \vee 1 \\ 1 \vee 0 & 1 \vee 0 & 0 \vee 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Integers and Matrices

- Assignment:

1. Why breaking down of large integer into set of small integers is preferred while performing integer arithmetic? Find sum of numbers 123,684 and 413,456 by representing the numbers as 4-tuple by using reminders modulo of pair-wise relatively prime numbers less than 100.

2. Find the value of x such that x = 1 (mod 3), x = 1 (mod 4), x = 1 (mod 5) and x = 0 (mod 7) using Chinese remainder theorem.

6. State Euclidean and extended Euclidean theorem. Write down extended Euclidean algorithm and illustrate it with example.

5. Find the value of x such that x = 1 (mod 5) and x = 2 (mod 7) using Chinese remainder theorem.

11. Define zero-one matrix. Explain the types of function. [1+4]

9. What does primality testing means? Describe how Fermat's Little Theorem tests for a prime number with suitable example.

Basic Discrete Structures

- References:
  - *Kenneth H. Rosen, Discrete mathematics and its applications, Seventh Edition McGraw Hill Publication, 2012.*