CS 3333: Mathematical Foundations

Modular Arithmetic

Modular Arithmetic

- Often times in computing, we are more concerned with what the remainder of an integer is when it is divided by some other integer than we are in the actual integer itself.

Modular Arithmetic

- Often times in computing, we are more concerned with what the remainder of an integer is when it is divided by some other integer than we are in the actual integer itself.
- For example, we might be interested in what time it will be 80 hours from now.

Modular Arithmetic

- Often times in computing, we are more concerned with what the remainder of an integer is when it is divided by some other integer than we are in the actual integer itself.
- For example, we might be interested in what time it will be 80 hours from now.
- We solve this by evaluating $14 + 80 \mod 24$ (14 is the current time, 80 is the amount of hours we are adding on, and 24 is the number of hours in a day).

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a-b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$?

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$? Does $3 \mid (2 - 5)$?

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$? Does $3 \mid (2 - 5)$? Yes.

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$? Does $3 \mid (2 - 5)$? Yes.
  - Is $17 \equiv 7 \pmod{5}$?

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$? Does $3 \mid (2 - 5)$? Yes.
  - Is $17 \equiv 7 \pmod{5}$? Does $5 \mid (17 - 7)$?

Modular Arithmetic

- **Definition**: Let a and b be integers and m be a positive integer. a is congruent to b modulo m if $m \mid (a - b)$.
- Notation:
  - $a \equiv b \pmod{m}$ if a is congruent to b modulo m.
  - $a \not\equiv b \pmod{m}$ if a is not congruent to b modulo m.
- Examples:
  - Is $2 \equiv 5 \pmod{3}$? Does $3 \mid (2 - 5)$? Yes.
  - Is $17 \equiv 7 \pmod{5}$? Does $5 \mid (17 - 7)$? Yes.

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".
- Example: Let $a = 6$, $b = 30$, and $m = 24$:

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24}$

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies 6 = 30 + k(24)$ for some int k $(k = -1)$.

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies 6 = 30 + k(24)$ for some int k $(k = -1)$.
  - $6 = 30 + -1 \cdot 24$

Modular Arithmetic

- **Theorem 4**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $a = b + km$ for some integer k.
- Note that when the claim is "if and only if" that one must prove the theorem in "both directions".
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies 6 = 30 + k(24)$ for some int k $(k = -1)$.
  - $6 = 30 + -1 \cdot 24 \implies 6 \equiv 30 \pmod{24}$.

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.
- Example: Let $a = 6$, $b = 30$, and $m = 24$:

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24}$

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies (6 \mod 24) = (30 \mod 24)$ (both are 6).

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies (6 \mod 24) = (30 \mod 24)$ (both are 6).
  - $(6 \mod 24) = (30 \mod 24)$

Modular Arithmetic

- **Theorem 3**: Let a and b be integers and m be a positive integer. $a \equiv b \pmod{m}$ if and only if $(a \mod m) = (b \mod m)$.
- Example: Let $a = 6$, $b = 30$, and $m = 24$:
  - $6 \equiv 30 \pmod{24} \implies (6 \mod 24) = (30 \mod 24)$ (both are 6).
  - $(6 \mod 24) = (30 \mod 24) \implies 6 \equiv 30 \pmod{24}$.

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16, 28

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16, 28, 40

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16, 28, 40, 52

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16, 28, 40, 52
- $4 + k \cdot 12$

Modular Arithmetic

- **Problem 26**: List 5 integers that are congruent to 4 modulo 12.
- 4, 16, 28, 40, 52
- $4 + k \cdot 12$
- In general, to find integers that are congruent to a modulo m:
  - $a + k \cdot m$ for any integer k.

Modular Arithmetic

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

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.
- $+_m$: $a +_m b = (a + b) \mod m$

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.
- $+_m$: $a +_m b = (a + b) \mod m$
- $\cdot_m$: $a \cdot_m b = a \cdot b \mod m$

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.
- $+_m$: $a +_m b = (a+b) \mod m$
- $\cdot_m$: $a \cdot_m b = a \cdot b \mod m$
- Examples $(a = 7, b = 9, m = 11)$:

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.
- $+_m$: $a +_m b = (a+b) \mod m$
- $\cdot_m$: $a \cdot_m b = a \cdot b \mod m$
- Examples ($a = 7, b = 9, m = 11$):
  - $7 +_{11} 9 = 16 \mod 11 = 5$.

Modular Arithmetic

- For any positive integer m, let $Z_m = \{0, 1, 2, \ldots, m-1\}$.
- Note that for any integer a, $(a \mod m) \in Z_m$.
- $+_m$: $a +_m b = (a+b) \mod m$
- $\cdot_m$: $a \cdot_m b = a \cdot b \mod m$
- Examples $(a = 7, b = 9, m = 11)$:
  - $7 +_{11} 9 = 16 \mod 11 = 5$.
  - $7 \cdot_{11} 9 = 63 \mod 11 = 8$.

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then
  - $a +_m b +_m c$

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then
  - $a +_m b +_m c = (a +_m b) +_m c$

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then
  - $a +_m b +_m c = (a +_m b) +_m c = a +_m (b +_m c)$
  - $a \cdot_m b \cdot_m c$

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then
  - $a +_m b +_m c = (a +_m b) +_m c = a +_m (b +_m c)$
  - $a \cdot_m b \cdot_m c = (a \cdot_m b) \cdot_m c$

Modular Arithmetic

- The $+_m$ and $\cdot_m$ operators satisfy several properties:
- **Closure**: If $a, b \in Z_m$, $a +_m b \in Z_m$ and $a \cdot_m b \in Z_m$.
- **Associativity**: If $a, b, c \in Z_m$ then
  - $a +_m b +_m c = (a +_m b) +_m c = a +_m (b +_m c)$
  - $a \cdot_m b \cdot_m c = (a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Modular Arithmetic

- **Commutativity**: If $a, b \in Z_m$ then
  - $a +_m b = b +_m a$
  - $a \cdot_m b = b \cdot_m a$

Modular Arithmetic

- **Commutativity**: If $a, b \in Z_m$ then
  - $a +_m b = b +_m a$
  - $a \cdot_m b = b \cdot_m a$
- **Distributivity**: If $a, b, c \in Z_m$ then
  - $(a +_m b) \cdot_m c = (a \cdot_m c) +_m (b \cdot_m c)$.

Modular Arithmetic

- **Identity Elements**:
  - $a +_m 0 = a \mod m$
  - $a \cdot_m 1 = a \mod m$

Modular Arithmetic

- **Identity Elements**:
  - $a +_m 0 = a \mod m$
  - $a \cdot_m 1 = a \mod m$
- **Additive Inverse**:
  - If $a \in Z_m$ then there exists a $b \in Z_m$ such that $a +_m b = 0$.

Modular Arithmetic

- **Identity Elements**:
  - $a +_m 0 = a \mod m$
  - $a \cdot_m 1 = a \mod m$
- **Additive Inverse**:
  - If $a \in Z_m$ then there exists a $b \in Z_m$ such that $a +_m b = 0$.
  - Example: if $m = 11$ and $a = 7$ then $b = 4$ ($(7 + 4) \mod 11 = 0$).

Modular Arithmetic

- Applications of Congruences (Section 4.5 in [KR]):
  - Hash Functions
  - Pseudorandom Numbers
  - Encryption/Decryption

Modular Arithmetic

- **Hash Functions**

Modular Arithmetic

- **Hash Functions**
  - Problem: We want to store information based off of some key/ID into memory, and we would like a quick way of storing and retrieving information. We could create an array whose size is the total number of possible keys, but this could require too much memory.

Modular Arithmetic

- **Hash Functions**
  - Problem: We want to store information based off of some key/ID into memory, and we would like a quick way of storing and retrieving information. We could create an array whose size is the total number of possible keys, but this could require too much memory.
  - Example: Suppose we want to store information about 100 UTSA students using their banner id as a key.
  - Solution: Create an array of size 100, and compute the location in the array to store the information by taking the banner id modulo 100.

Modular Arithmetic

- **Hash Functions**
  - Problem: We want to store information based off of some key/ID into memory, and we would like a quick way of storing and retrieving information. We could create an array whose size is the total number of possible keys, but this could require too much memory.
  - Example: Suppose we want to store information about 100 UTSA students using their banner id as a key.
  - Solution: Create an array of size 100, and compute the location in the array to store the information by taking the banner id modulo 100.
  - For a student with banner id 00687581, we would store information in location $00687581 \mod 100 = 81$.

Modular Arithmetic

- **Pseudorandom Numbers**

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.
  - We would like to compute a deterministic series of numbers which appear to be random.

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.
  - We would like to compute a deterministic series of numbers which appear to be random.
- Linear Congruential Method:

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.
  - We would like to compute a deterministic series of numbers which appear to be random.
- Linear Congruential Method:
  - To compute random numbers between 0 and m, solve the following recursive function:

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.
  - We would like to compute a deterministic series of numbers which appear to be random.
- Linear Congruential Method:
  - To compute random numbers between 0 and m, solve the following recursive function:
  - $x_{n+1} = (a \cdot x_n + c) \mod m$ where a and c are constants.

Modular Arithmetic

- **Pseudorandom Numbers**
  - Computers cannot simply generate random numbers on their own.
  - We would like to compute a deterministic series of numbers which appear to be random.
- Linear Congruential Method:
  - To compute random numbers between 0 and m, solve the following recursive function:
  - $x_{n+1} = (a \cdot x_n + c) \mod m$ where a and c are constants.
  - Often times, x_0 (the *seed*) is the system time $\mod m$.

Modular Arithmetic

- **Encryption/Decryption**

Modular Arithmetic

- **Encryption/Decryption**
  - Suppose we want to send a message written with capital letters A to Z. We can encrypt the message by replacing each letter with the letter "three positions" to the right. For X, Y, and Z, we use A, B, and C respectively.

Modular Arithmetic

- **Encryption/Decryption**
  - Suppose we want to send a message written with capital letters A to Z. We can encrypt the message by replacing each letter with the letter "three positions" to the right. For X, Y, and Z, we use A, B, and C respectively.
  - Encrypting: $(p + 3) \mod 26$ where p is the position of the current letter before encrypting.

Modular Arithmetic

- **Encryption/Decryption**
  - Suppose we want to send a message written with capital letters A to Z. We can encrypt the message by replacing each letter with the letter "three positions" to the right. For X, Y, and Z, we use A, B, and C respectively.
  - Encrypting: $(p + 3) \mod 26$ where p is the position of the current letter before encrypting.
  - Decrypting: $(q - 3) \mod 26$ where q is the position of the current letter before decrypting.