CSE2023 Discrete Computational Structures

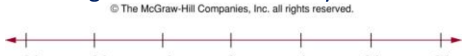
Lecture 12

4.1 Divisibility and modular arithmetic

- **Number theory**: the branch of mathematics involves integers and their properties
- If ***a*** and ***b*** are integers with **a≠0**, we say that ***a*** divides ***b*** if there is an integer c s.t. ***b=ac***
- When **a** divides **b** we say that **a** is a **factor** of **b** and that **b** is a **multiple** of **a**
- The notation **a | b** denotes **a** divides **b**. We write **a** $\nmid$ **b** when does not divide **b**

Example

- Let **n** and **d** be positive integers. How many positive integers not exceeding **n** are divisible by **d**?
- The positive integers divisible by d are all integers of them form **dk**, where **k** is a positive integer
- Thus, there are $\lfloor n/d \rfloor$ positive integers not exceeding **n** that are divisible by **d**



Theorem and corollary

- Theorem: Let a, b, and c be integers, then
  - If a | b and a| c, then a | (b+c)
  - If a | b, and a | bc for all integers c
  - If a | b and b | c, then a | c
- Corollary: If a, b, and c are integers s.t. a | b and a | c, then a | mb+nc whenever m and n are integers

The division algorithm

- Let **a** be integer and **d** be a positive integer. Then there are unique integers **q** and **r** with **0 ≤ r < d**,
  s.t. **a=dq+r**
- In the equality, **q** is the **quotient**, **r** is the **remainder**
  q = a **div** d, r = a **mod** d
- -11 divided by 3
- -11=3(-4)+1, -4=-11 div 3, **1=-11 mod 3**
- -11=3(-3)-2, but **remainder cannot be negative**

5

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≡b** (mod m) to indicate that **a** is **congruent** to **b** modulo **m**
- If **a** and **b** are not congruent modulo **m**, we write a ≢b (mod m)
- Let **a** and **b** be integers, **m** be a positive integer. Then **a≡b** (mod **m**) if and only if **a** mod **m = b** mod **m**

6

Example

- Determine whether 17 is congruent to 5 modulo 6, and whether 24 and 14 are not congruent modulo 6
  - 17-5=12, we see 17≡5 (mod 6)
  - 24-14=10, and thus 24≢14 (mod 6)

7

Theorem

- Karl Friedrich Gauss developed the concept of congruences at the end of 18^{th} century
- Let **m** be a positive integer. The integer **a** and **b** are congruent modulo **m** if and only if there is an integer **k** such that **a=b+km**
  - (→) If a=b+km, then km=a-b, and thus m divides a-b and so a≡b (mod m)
  - (←) if a≡b (mod m), then m | a-b. Thus, a-b=km, and so a=b+km

8

Theorem

- Let **m** be a positive integer. If a ≡ b (mod m) and c ≡ d (mod m), then a+c=b+d (mod m) and ac ≡ bd (mod m)
  - Since a ≡ b (mod m) and c ≡ d (mod m), there are integers s.t. b=a+sm and d=c+tm
  - Hence, b+d=(a+c)+m(s+t), bd=(a+sm)(c+tm)=ac+m(at+cs+stm)
  - Hence a+c ≡ b+d (mod m), and ac ≡ bd (mod m)

Example

- 7 ≡ 2 (mod 5) and 11 ≡ 1 (mod 5), so
  - 18=7+11 ≡ 2+1=3 (mod 5)
  - 77=7·11 ≡2·1=2(mod 5)

4.2 Integer representations and algorithms

- Base **b expansion** of **n**
- For instance, $(245)_8=2*8^2+4*8+5=165$
- Hexadecimal expansion of (2AE0B)16

  $(2AE0B)_{16}=2*16^4+10*16^3+14*16^2+0*16+11=175627$
- Constructing **base b expansion**

Base conversion

- Constructing the base b expansion

  $n=bq_0+a_0$, $0 \le a_0<b$
- The remainder a_0, is the rightmost digit in the base b expansion of n
- Next, divide q_0 by b to obtain

  $q_0=bq_1+a_1$, $0\le a_1<b$
- We see 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, until the quotient is zero

Example

- Find the octal base of $(12345)_{10}$
- First, 12345=8*1543+1
- Successively dividing quotients by 8 gives

  1543=8*192+7

  192=8*24+0

  24=8*3+0

  3=8*0+3
- $(12345)_{10}=(30071)_8$

Modular exponentiation

- Need to find **b^n mod m** efficiently in cryptography
- Impractical to compute b^n and then mod m
- Instead, find binary expansion of n first, e.g., $n=(a_{k-1} \dots a_1\ a_0)$

$$b^n = b^{a_{k-1}\cdot 2^{k-1}+\cdots+a_1\cdot 2+a_0} = b^{a_{k-1}\cdot 2^{k-1}} b^{a_{k-2}\cdot 2^{k-2}} \dots b^{a_1\cdot 2} b^{a_0}$$

- To compute b^n , first find the values of b, b^2, …, $(b^4)^2=b^8$, …
- Next multiple the b^{2^j} where $a_j=1$

Example

- To compute 3^{11}
- $11=(1011)_2$,So $3^{11}=3^8\ 3^2\ 3^1$. First compute $3^2=9$, and then $3^4=9^2=81$, and $3^8=(3^4)^2=(81)^2=6561$, So $3^{11}=6561*9*3=177147$
- The algorithm successively finds b mod m, b^2 mod m, b^4 mod m, …, $b^{2^{k-1}}$ mod m, and multiply together those terms

Algorithm

- **procedure** *modular exponentiation* (***b****:integer,* ***n****=($a_{k-1}a_{k-2} \dots a_1a_0$, …, a_n)$_2$,* **m**:positive integer)

  x := 1

  power:=b mod m

  for i:=0 to k-1

  if a_i =1 **then** x:=(x· power) mod m

  power:=(power·power) mod m

  end

  {x equals b^n mod m}
- *It uses $O((\log m)^2$ long n) bit operations*

Example

- Compute 3^{644} mod 645
  - First note that $644=(1010000100)_2$
  - At the beginning, x=1, power=3 mod 645 = 3
  - i=0, a_0=0, x=1, power=3^2 mod 645=9
  - i=1, a_1=0, x=1, power=9^2 mod 645=81
  - **i=2, a_2=1, x=(1*81) mod 645=81, power=81^2 mod 645=6561 mod 645=111**
  - i=3, a_3=0, x=81, power=111^2 mod 645=12321 mod 645=66
  - i=4, a_4=0, x=81, power=66^2 mod 645=4356 mod 645=486
  - i=5, a_5=0, x=81, power=486^2 mod 645=236196 mod 645=126
  - i=6, a_6=0, x=81, power=126^2 mod 645=15876 mod 645=396
  - **i=7, a_7=1, x=(81*396) mod 645=471, power=396^2 mod 645=156816 mod 645=81**
  - i=8, a_8=0, x=471, power=81^2 mod 645=6561mod 645=111
  - **i=9, a_9=1, x=(471*111) mod 645=36**
- 3^{644} mod 645=36

4.3 Primes and greatest common divisions

- **Prime**: a positive integer **p** greater than 1 if the only positive factors of **p** are **1** and **p**
- A positive integer greater than 1 that is not prime is called **composite**
- **Fundamental theorem of arithmetic**: Every positive integer greater than 1 can be written uniquely as a prime or as the product of two or more primes when the prime factors are written in order of non-decreasing size

Example

- Prime factorizations of integers
  - $100=2\cdot2\cdot5\cdot5=2^2\cdot5^2$
  - 641=641
  - $999=3\cdot3\cdot3\cdot37=3^3\cdot37$
  - $1024=2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2\cdot2=2^{10}$

Theorem

- Theorem: If **n** is a composite integer, then **n** has a prime division less than or equal to $\sqrt{n}$
- As **n** is composite, **n** has a factor 1<a<n, and thus n=ab
- We show that a≤$\sqrt{n}$ or b ≤$\sqrt{n}$ (by contraposition)
- Thus n has a divisor not exceeding $\sqrt{n}$
- This divisor is either prime or by the fundamental theorem of arithmetic, **has a prime divisor less than itself**, and thus a prime divisor less than less than $\sqrt{n}$
- In either case, n has a prime divisor b ≤ $\sqrt{n}$

Example

- Show that 101 is prime
- The only primes not exceeding $\sqrt{101}$ are 2, 3, 5, 7
- As 101 is not divisible by 2, 3, 5, 7, it follows that 101 is prime

Procedure for prime factorization

- Begin by diving **n** by successive primes, starting with **2**
- If **n** has a prime factor, we would find a prime factor not exceeding $\sqrt{n}$
- If no prime factor is found, then **n** is prime
- Otherwise, if a prime factor p is found, continue by factoring n/p
- Note that n/p has no prime factors less than p
- If n/p has no prime factor greater than or equal to p and not exceeding its square root, then it is prime
- Otherwise, if it has a prime factor q, continue by factoring n/(pq)
- Continue until factorization has been reduced to a prime

Example

- Find the prime factorization of 7007
- Start with 2, 3, 5, and then 7, 7007/7=1001
- Then, divide 1001 by successive primes, beginning with 7, and find 1001/7=143
- Continue by dividing 143 by successive primes, starting with 7, and find 143/11=13
- As 13 is prime, the procedure stops
- $7007=7\cdot 7\cdot 11\cdot 13=7^2\cdot 11\cdot 13$