Information Security System
EC-615-F

Lecture no 2, 3,4,5,6

Topics Covered

- Finite field of increasing importance in cryptography
  - AES, Elliptic Curve, CMAC
- concern operations on “numbers”
  - where what constitutes a “number” and the type of operations varies considerably
- start with basic number theory concepts
  - divisibility, Euclidian algorithm, modular arithmetic

Divisors

- say a non-zero number `b` **divides** `a` if for some `m` have `a=mb` (`a,b,m` all integers)
- that is `b` divides into `a` with no remainder
- denote this `b|a`
- and say that `b` is a **divisor** of `a`
- eg. all of 1,2,3,4,6,8,12,24 divide 24
- eg. 13 | 182; –5 | 30; 17 | 289; –3 | 33; 17 | 0

Properties of Divisibility

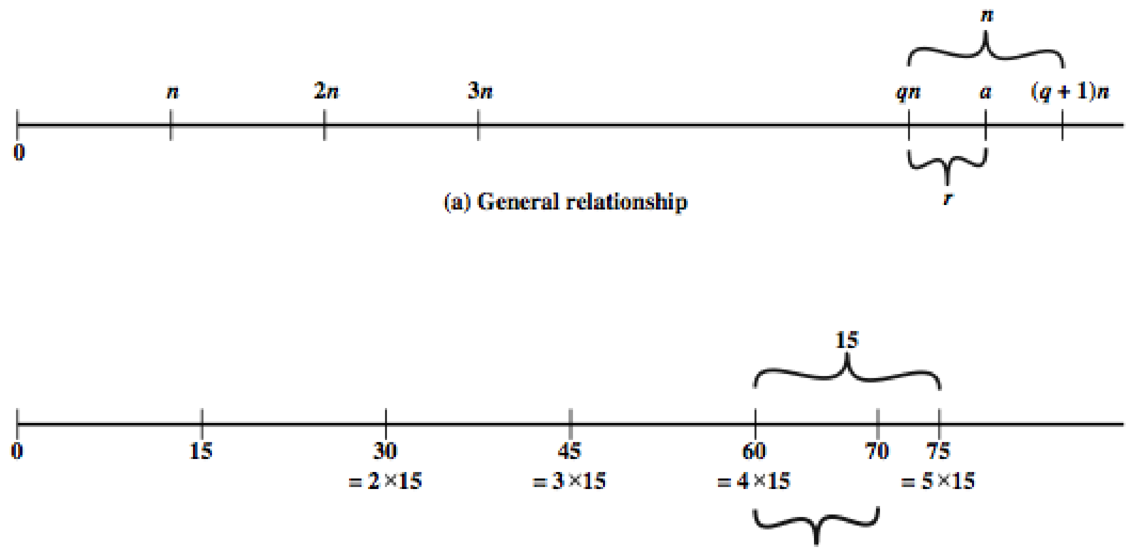
- If $a|1$, then $a = \pm 1$.
- If $a|b$ and $b|a$, then $a = \pm b$.
- Any b != 0 divides 0.
- If $a \mid b$ and $b \mid c$, then $a \mid c$
  - e.g. $11 \mid 66$ and $66 \mid 198 \rightarrow 11 \mid 198$
- If $b|g$ and $b|h$, then $b|(mg + nh)$

  for arbitrary integers m and n

  e.g. $b = 7$; $g = 14$; $h = 63$; $m = 3$; $n = 2$

  hence $7|14$ and $7|63 \rightarrow 7|(3 \times 14 + 2 \times 63)$

Division Algorithm



(a) General relationship

(b) Example: 70 = (4×15) + 10

Greatest Common Divisor (GCD)

- a common problem in number theory
- gcd (a,b) of a and b is the largest integer that divides both a and b
  - E.g., gcd(60,24) = 12
- define gcd(0, 0) = 0, gcd(a,0) = |a| for a !=0
- often want **no common factors** (except 1) define such numbers as **relatively prime**
  - E.g. gcd(8,15) = 1
  - hence 8 & 15 are relatively prime

Euclidean Algorithm

- A simple procedure for finding d = gcd (a, b)
- gcd(|a|, |b|) = gcd (a, b) = gcd (b, a)
- no harm to assume a >= b > 0

- ```
  Euclid(a,b)
      if (b=0) then return a;
      else return Euclid(b, a mod b);
  ```

- E.g., gcd(60,24) = 12; gcd(8,15) = 1
  ```

Example GCD(1970,1066)

```
1970 = 1 x 1066 + 904    gcd(1066, 904)
1066 = 1 x 904 + 162  gcd(904, 162)
904 = 5 x 162 + 94    gcd(162, 94)
162 = 1 x 94 + 68     gcd(94, 68)
94 = 1 x 68 + 26      gcd(68, 26)
68 = 2 x 26 + 16      gcd(26, 16)
26 = 1 x 16 + 10      gcd(16, 10)
16 = 1 x 10 + 6     gcd(10, 6)
10 = 1 x 6 + 4      gcd(6, 4)
6 = 1 x 4 + 2       gcd(4, 2)
4 = 2 x 2 + 0       gcd(2, 0)
```

GCD(1160718174, 316258250)

Dividend	Divisor	Quotient	Remainder
a = 1160718174	b = 316258250	q1 = 3	r1 = 211943424
b = 316258250	r1 = 211943424	q2 = 1	r2 = 104314826
r1 = 211943424	r2 = 104314826	q3 = 2	r3 = 3313772
r2 = 104314826	r3 = 3313772	q4 = 31	r4 = 1587894
r3 = 3313772	r4 = 1587894	q5 = 2	r5 = 137984
r4 = 1587894	r5 = 137984	q6 = 11	r6 = 70070
r5 = 137984	r6 = 70070	q7 = 1	r7 = 67914
r6 = 70070	r7 = 67914	q8 = 1	r8 = 2516
r7 = 67914	r8 = 2516	q9 = 31	r9 = 1078
r8 = 2516	r9 = 1078	q10 = 2	r10 = 0

There are other GCD algorithms, but Euclidean Algorithm is very efficient!

Modular Arithmetic

- define **modulo operator** "`a mod n`" to be remainder when a is divided by n
  - where positive integer *n* is called the **modulus**
  - `a = qn + r        0<=r<n; q=floor(a/n)`
  - `a = floor(a/n) * n + (a mod n)`
  - `e.g, 11 mod 7 = 4;    -11 mod 7 = 3`
- *a* and *b* are **congruent modulo n** if: `a mod n = b mod n`
  - when divided by *n*, a & b have same remainder
  - a ≡ b (mod n),   eg. 100 ≡ 34 mod 11

Modular Arithmetic Operations

- (mod n) operator maps all integers into the set

  $Z_n = \{0, 1, \ldots, (n-1)\}$

- can perform arithmetic operations within the confines of this set → modular arithmetic

- Rules for addition, subtraction, and multiplication carry over into modular arithmetic

Properties of Modular Arithmetic Operations

1. [(a mod n) + (b mod n)] mod n = (a + b) mod n
2. [(a mod n) – (b mod n)] mod n = (a – b) mod n
3. [(a mod n) x (b mod n)] mod n = (a x b) mod n

e.g.

[(11 mod 8) + (15 mod 8)] mod 8 = 10 mod 8 = 2 (11 + 15) mod 8 = 26 mod 8 = 2

[(11 mod 8) – (15 mod 8)] mod 8 = –4 mod 8 = 4 (11 – 15) mod 8 = –4 mod 8 = 4

[(11 mod 8) x (15 mod 8)] mod 8 = 21 mod 8 = 5 (11 x 15) mod 8 = 165 mod 8 = 5

Modulo 8 Addition in Z_8

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

The matrix is symmetric about the main diagonal

Additive inverse exists to each integer in modular addition:
$(x+y) \bmod 8 = 0$

Modulo 8 Multiplication in Z_8

+	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

The matrix is symmetric about the main diagonal

Multiplicative inverse exists to some integers in mod 8 multiplication:
(x * y) mod 8 = 1

Residue Classes (mod n)

- (mod n) operator maps all integers into the set

  $Z_n = \{0, 1, . . . , (n - 1)\}$ → *set of residues, or residue classes*

- Each integer in Z_n represents a residue class

  $[r] = \{a: a \text{ is an integer}, a \equiv r \pmod{n}\}$

  e.g., the residue classes (mod 4) are:

  $[0] = \{\ldots, -16, -12, -8, -4, 0, 4, 8, 12, 16, \ldots\}$

  $[1] = \{\ldots, -15, -11, -7, -3, 1, 5, 9, 13, 17, \ldots\}$

  $[2] = \{\ldots, -14, -10, -6, -2, 2, 6, 10, 14, 18, \ldots\}$

  $[3] = \{\ldots, -13, -9, -5, -1, 3, 7, 11, 15, 19, \ldots\}$

- Finding the smallest nonnegative integer to which k is congruent modulo n is called **reducing k modulo n**

Properties of Modular Arithmetic for Integers in Z_n

Property	Expression
Commutative laws	$(w + x) \bmod n = (x + w) \bmod n$ $(w \times x) \bmod n = (x \times w) \bmod n$
Associative laws	$[(w + x) + y] \bmod n = [w + (x + y)] \bmod n$ $[(w \times x) \times y] \bmod n = [w \times (x \times y)] \bmod n$
Distributive law	$[w \times (x + y)] \bmod n = [(w \times x) + (w \times y)] \bmod n$
Identities	$(0 + w) \bmod n = w \bmod n$ $(1 \times w) \bmod n = w \bmod n$
Additive inverse $(-w)$	For each $w \in Z_n$, there exists a z such that $w + z \equiv 0 \bmod n$

Modular Arithmetic Special Properties

- if (a + b) ≡ (a + c) (mod n) then b ≡ c (mod n)
  - e.g., (5 + 23) ≡ (5 + 7) (mod 8) → 23 ≡ 7(mod 8)
  - due to the existence of additive inverse
  - add additive inverse –a to both sides to prove

- if (a * b) ≡ (a * c) (mod n) then b ≡ c (mod n) if a is relatively prime to n
  - e.g., (5 * 23) ≡ (5 * 7) (mod 8) → 23 ≡ 7(mod 8)
  - if multiplicative inverse exists for a mod n
  - normally, if an integer is relatively prime to n, then this integer has a multiplicative inverse in Z_n

Extended Euclidean Algorithm

- calculates not only GCD but x & y (with opposite signs): ax + by = d = gcd(a, b)
- useful for later crypto computations, e.g, RSA
- follow sequence of divisions for GCD but assume at each step i, can find x & y:

  r = ax + by
- at end find GCD value and also x & y