Chapter 2

Introduction to Number Theory

Outline

- Divisibility and the division algorithm
- The Euclidean algorithm
  - Greatest Common Divisor
- Modular arithmetic
  - The modulus
  - Properties of congruences
  - Modular arithmetic operations
  - Properties of modular arithmetic
  - Euclidean algorithm revisited
  - The extended Euclidean algorithm
- Prime numbers
- Fermat's Theorem
- Euler's totient function
- Euler's Theorem
- Testing for primality
- The Chinese Remainder Theorem
- Discrete logarithms
  - Powers of an integer, modulo *n*
  - Logarithms for modular arithmetic
  - Calculation of discrete logarithms

Divisibility

- We say that a nonzero b **divides** a if $a = mb$ for some m, where a, b, and m are integers
- b divides a if there is no remainder on division
- The notation $b \mid a$ is commonly used to mean b divides a
- If $b \mid a$ we say that b is a **divisor** of a

The positive divisors of 24 are 1, 2, 3, 4, 6, 8, 12, and 24
$13 \mid 182;\ -5 \mid 30;\ 17 \mid 289;\ -3 \mid 33;\ 17 \mid 0$

Properties of Divisibility

- If $a \mid 1$, then $a = \pm 1$
- If $a \mid b$ and $b \mid a$, then $a = \pm b$
- Any $b \neq 0$, $b \mid 0$
- If $a \mid b$ and $b \mid c$, then $a \mid c$

$$\boxed{11 \mid 66 \text{ and } 66 \mid 198 = 11 \mid 198}$$

- If $b \mid g$ and $b \mid h$, then $b \mid (mg + nh)$ for arbitrary integers m and n

Properties of Divisibility

- To see this last point, note that:
  - If $b \mid g$, then g is of the form $g = b * g_1$ for some integer g_1
  - If $b \mid h$, then h is of the form $h = b * h_1$ for some integer h_1
- So:
  - $mg + nh = mbg_1 + nbh_1 = b * (mg_1 + nh_1)$

    and therefore b divides $mg + nh$

> $b = 7$; $g = 14$; $h = 63$; $m = 3$; $n = 2$
> $7 \mid 14$ and $7 \mid 63$.
> To show 7 (3 * 14 + 2 * 63),
> we have (3 * 14 + 2 * 63) = 7(3 * 2 + 2 * 9),
> and it is obvious that $7 \mid (7(3 * 2 + 2 * 9))$.

Division Algorithm

- Given any positive integer **n** and any nonnegative integer ***a***, if we divide ***a*** by **n** we get an integer quotient ***q*** and an integer remainder ***r*** that obey the following relationship:

  $a = q \times n + r$

  where:

  - $0 \le r < n$
  - $q = \lfloor a/n \rfloor$

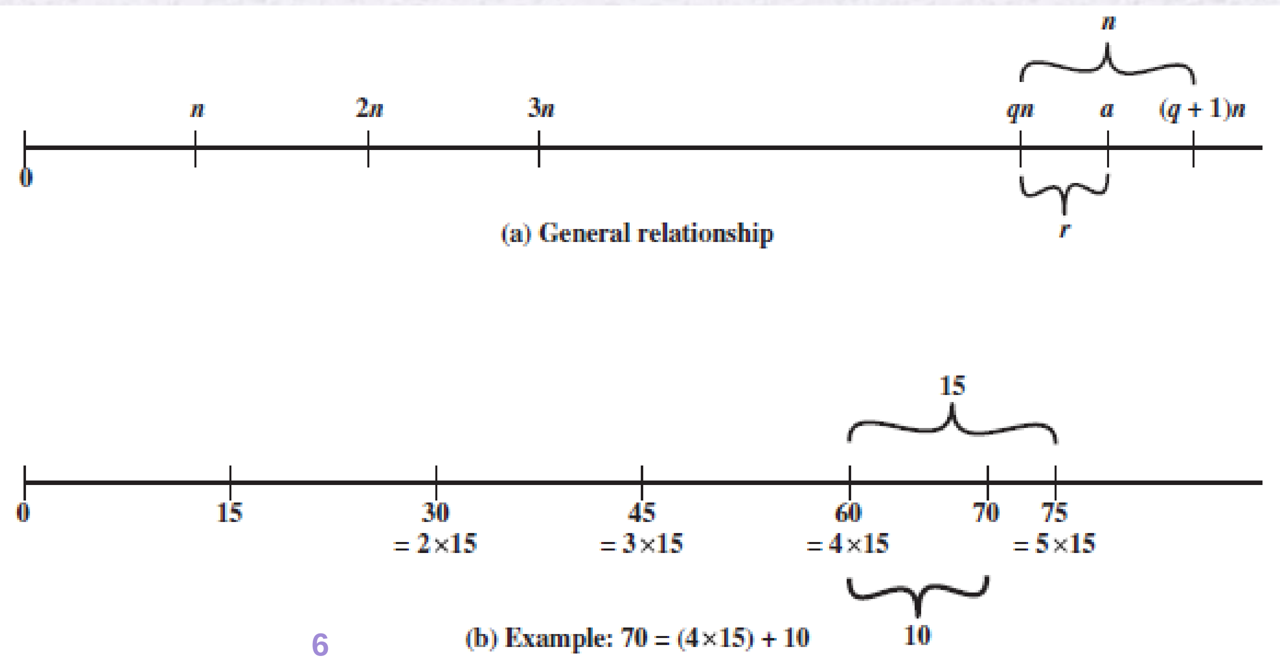

Figure 4.1 The Relationship $a = qn + r; 0 \le r < n$

Euclidean Algorithm

- Procedure for determining the greatest common divisor of two positive integers
- Two integers are **relatively prime** if their only common positive integer factor is 1

Greatest Common Divisor (GCD)

- The greatest common divisor of *a* and *b* is the largest integer that divides both *a* and *b*
- We can use the notation gcd(*a*,*b*) to mean the **greatest common divisor** of *a* and *b*
- We also define gcd(0,0) = 0
- Positive integer c is said to be the gcd of *a* and *b* if:
  - c is a divisor of *a* and *b*
  - Any divisor of *a* and *b* is a divisor of c
- An equivalent definition is:

$$\gcd(a,b) = \max[k, \text{ such that } k \mid a \text{ and } k \mid b]$$

GCD

- Because we require that the greatest common divisor be positive, gcd(a,b) = gcd(a,$-b$) = gcd($-a$,b) = gcd($-a$,$-b$)
- In general, gcd(a,b) = gcd($|a|$, $|b|$)

$$\boxed{\gcd(60, 24) = \gcd(60, -24) = 12}$$

- Also, because all nonzero integers divide 0: $\boxed{\gcd(a,0) = |a|}$
- We stated that two integers *a* and *b* are relatively prime if their only common positive integer factor is 1;
- **⇒ *a* and *b* are relatively prime if gcd(*a*,*b*) = 1**

8 and 15 are relatively prime because the positive divisors of 8 are 1, 2, 4, and 8, and the positive divisors of 15 are 1, 3, 5, and 15. So 1 is the only integer on both lists.

Methods to Compute GCD

- GCD is computed using different methods
  1. Division
  2. Modulus
  3. Subtraction based
- Comparison
  - Division and Modulus are very similar, they take **less steps,** but steps are more complex compared with subtraction.
  - Subtraction is easy but takes larger number of steps.

Compute GCD using Division

// Compute GCD (a | b)
// using division

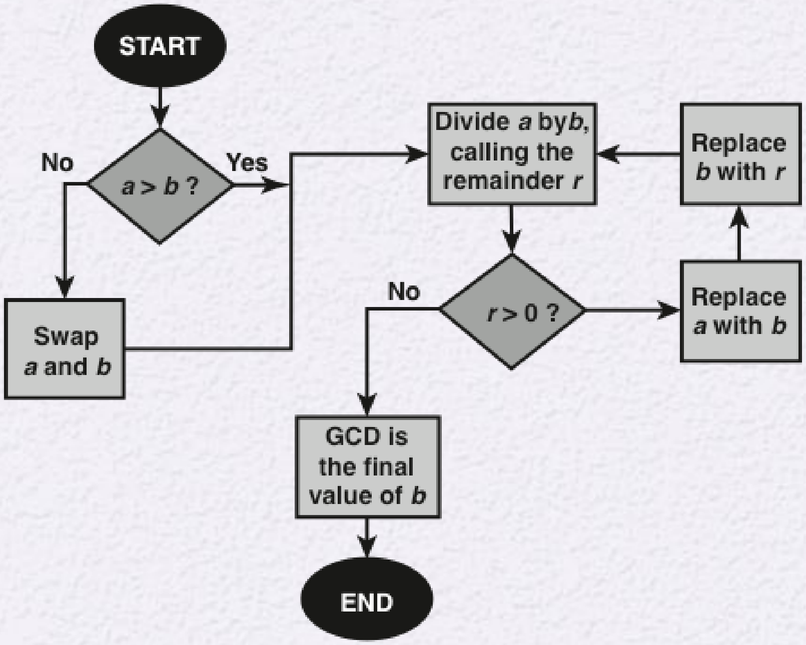

Figure 2.2 Euclidean Algorithm

Example: GCD (710, 310)=10

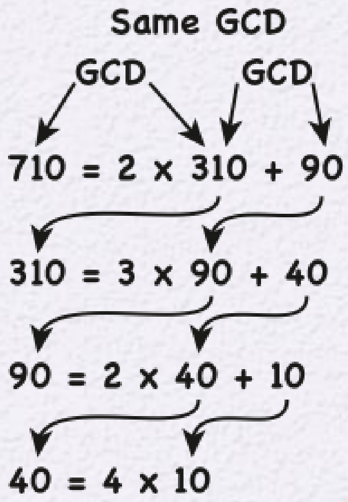

Figure 2.3 Euclidean Algorithm Example: gcd(710, 310)

Another Example : GCD using Division

GCD (1160718174 , 316258250) = 1078

Dividend	Divisor	Quotient	Remainder
a = 1160718174	b = 316258250	q_1 = 3	r_1 = 211943424
b = 316258250	r_1 = 211943424	q_2 = 1	r_2 = 104314826
r_1 = 211943424	r_2 = 104314826	q_3 = 2	r_3 = 3313772
r_2 = 104314826	r_3 = 3313772	q_4 = 31	r_4 = 1587894
r_3 = 3313772	r_4 = 1587894	q_5 = 2	r_5 = 137984
r_4 = 1587894	r_5 = 137984	q_6 = 11	r_6 = 70070
r_5 = 137984	r_6 = 70070	q_7 = 1	r_7 = 67914
r_6 = 70070	r_7 = 67914	q_8 = 1	r_8 = 2156
r_7 = 67914	r_8 = 2156	q_9 = 31	r_9 = 1078
r_8 = 2156	r_9 = 1078	q_{10} = 2	r_{10} = 0

Compute GCD using Modulus

```
// Compute  GCD (a , b )
// using modulus operation

if (b > a)  {
   temp=a;
   a=b;
   b=temp;
 }

while  (r != 0)  {
  // a = b * k + r
   r = a % b;
   a = b;
   b = r;
 }

GCD = a;
```

GCD (710, 310)=10

a	b	r= a mod b
710	310	90
310	90	40
90	40	10
40	10	0

Compute GCD using Subtraction

```
// Compute  GCD (a , b )
// using modulus operation


while  (a != b)  {
    if  (a < b)
       b = b - a;
   else
       a = a - b;
 }

GCD = a;
```

Example: GCD (710, 310)=10

a	b	|a-b|
710	310	400
400	310	90
90	310	220
90	220	130
90	130	40
90	40	50
50	40	10
10	40	30
10	30	20
10	20	10
10	10	0

Modular Arithmetic

- The modulus
  - If *a* is an integer and ***n* is a positive integer**, we define ***a* mod *n*** to be the remainder when *a* is divided by *n*; the integer *n* is called the **modulus**
  - Thus, for any integer *a:*

$$a = qn + r \qquad 0 \le r < n;\ q = \lfloor a/n \rfloor$$

$$a = \lfloor a/n \rfloor * n + (a \bmod n)$$

$\lfloor x \rfloor$: *floor operation is the largest integer less than or equal to x*

- Examples of floor operation
  - $\lfloor 2.3 \rfloor = 2.0$
  - $\lfloor -2.3 \rfloor = -3$
- Example of modulus:
  - 11 mod 7 = 4

Examples of Modulus Operations (for positive and negative values)

- To compute modulus of positive/negative numbers apply:

$$a = \lfloor a/n \rfloor \times n + (a \bmod n)$$

Example	Explanation
11 (mod 7) = 4	$11 = \lfloor 11/7 \rfloor \times 7 + (11 \bmod 7)$ $11 = 1 \times 7 + (11 \bmod 7)$ $11 = 7 + (11 \bmod 7)$
-11 (mod 7) = 3	$-11 = \lfloor -11/7 \rfloor \times 7 + (-11 \bmod 7)$ $-11 = -2 \times 7 + (-11 \bmod 7)$ $-11 = -14 + (-11 \bmod 7)$
11 (mod -7) = -3	$11 = \lfloor 11/-7 \rfloor \times -7 + (11 \bmod -7)$ $11 = -2 \times -7 + (11 \bmod -7)$ $11 = 14 + (11 \bmod -7)$
-11 (mod -7) = -4	$-11 = \lfloor -11/-7 \rfloor \times -7 + (-11 \bmod -7)$ $-11 = 1 \times -7 + (-11 \bmod -7)$ $-11 = -7 + (-11 \bmod -7)$

In general, ***n*** should be positive

Modular Arithmetic

- Congruent modulo n
  - Two integers a and b are said to be **congruent modulo** $\boldsymbol{n}$ if (a mod n) = (b mod n)
  - This is written as $a \equiv b \pmod{n}$
  - Note that if $a = 0 \pmod{n}$, then $n \mid a$

$$73 \equiv 4 \pmod{23}; \qquad 21 \equiv -9 \pmod{10}$$

Properties of Congruences

- Congruences have the following properties:

  1. $a \equiv b \pmod{n}$ if $n | (a - b)$

  2. $a \equiv b \pmod{n}$ implies $b \equiv a \pmod{n}$

  3. $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ imply $a \equiv c \pmod{n}$

- To demonstrate the first point, if $n | (a - b)$, then $(a - b) = kn$ for some k
  - So we can write $a = b + kn$
  - Therefore, (a mod n) = (remainder when $b + kn$ is divided by n) = (remainder when b is divided by n) = (b mod n)

$23 \equiv 8 \pmod{5}$ because $23 - 8 = 15 = 5 * 3$
$-11 \equiv 5 \pmod{8}$ because $-11 - 5 = -16 = 8 * (-2)$
$81 \equiv 0 \pmod{27}$ because $81 - 0 = 81 = 27 * 3$

Modular Arithmetic

- Modular arithmetic exhibits the following properties:
  1. $[(a \bmod n) + (b \bmod n)] \bmod n = (a + b) \bmod n$
  2. $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$
  3. $[(a \bmod n) * (b \bmod n)] \bmod n = (a * b) \bmod n$
- We demonstrate the first property:
  - Define $(a \bmod n) = r_a$ and $(b \bmod n) = r_b$. Then we can write
    - $a = r_a + jn$ for some integer j
    - $b = r_b + kn$ for some integer k
  - Then:

$$\begin{aligned}(a + b) \bmod n &= (r_a + jn + r_b + kn) \bmod n \\ &= (r_a + r_b + (k + j)n) \bmod n \\ &= (r_a + r_b) \bmod n \\ &= [(a \bmod n) + (b \bmod n)] \bmod n\end{aligned}$$

Advantage of modular arithmetic

- The main advantage of modular arithmetic (properties) is to simplify evaluating large number in multiplication and addition operations. Consider the following example.

- Example: **Evaluate (1425 * 3964 * 7899 * 5501) mod 13**

(1425 * 3964 * 7899 * 5501) mod 13
= 245,449,566,231,300 mod 13
= 2

Note: 245,449,566,231,300 is a very large number.

- Using modular arithmetic properties:

(1425 * 3964 * 7899 * 5501) mod 13
= [(1425 mod 13) * (3964 mod 13) * (7899 mod 13) * (5501 mod 13)] mod 13
= [8 * 12 * 8 * 2] mod 13
= 1536 mod 13
= 2

Examples of Remaining Properties:

- Examples of the three remaining properties:

11 mod 8 = 3; 15 mod 8 = 7

[(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) * (15 mod 8)] mod 8 = 21 mod 8 = 5

(11 * 15) mod 8 = 165 mod 8 = 5

Exponentiation and Modulus

- Exponentiation is performed by repeated multiplication, as in ordinary arithmetic

To find 11^7 mod 13, we can proceed as follows:

$$11^2 = 121 \equiv 4 \pmod{13}$$
$$11^4 = (11^2)^2 \equiv 4^2 \equiv 3 \pmod{13}$$
$$11^7 \equiv 11 \times 4 \times 3 \equiv 132 \equiv 2 \pmod{13}$$

Compute: 101^{1001} mod 7

$3 \equiv 101$ mod 7

$3^{1000} = [[3^{10}]^{10}]^{10}$ mod7

$= [4^{10}]^{10}$ mod7

$= 4^{10}$ mod7

$= 4$

$3^{1001} = 4 * 3$ mod7 $= 5$

Compute: $100{,}001^{100{,}001}$ mod 19

Answer: 16

Compute: 1234^{2002} mod 11

Answer: 4

Addition Modulo 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

As we move down the tables, rows are rotated to left.

(This table can be found on page 37 in the textbook)

Multiplication Modulo 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

Multiplication modulo is not as predictive as addition Modulo

(This table can be found on page 37 in the textbook)

Additive and Multiplicative Inverse Modulo 8

- **additive inverse or negative (-w) :**

x is negative of y if
(x + y) mod 8 = 0

- **Multiplicative inverse (w^{-1}) :**

x is multiplicative invers of y if
(x × y) mod 8 = 1

w	$-w$	w^{-1}
0	0	—
1	7	1
2	6	—
3	5	3
4	4	—
5	3	5
6	2	—
7	1	7

+	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

×	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

Why Inverses are Important in Cryptography

- Inverse supports decryption calculations.
- Consider the following example.
- We desire to encrypt message (M=11), key (K=12) and produce cipher (c). We will use (mod 17) function.

	Example of Modulo Addition	Example of Modulo Multiplication
Encryption	$C=(K+M) \bmod 17$	$C=(K\times M) \bmod 17$
M	11	11
K	12	12
Inverse	$-K = 5$	$K^{-1} = 10$
Encryption	$C=(K+M) \bmod 17 = 6$	$C=(K\times M) \bmod 17 = 13$
Decryption	$M=(-K + C) = (5+6) \bmod 17 = 11 =$ **M**	$M=(K^{-1}\times C) = (10\times 13) \bmod 17 = 11=$ **M**

Integers in Z_n

- Define the set Z_n as the set of nonnegative integers less than *n*: $Z_n = \{0, 1, ..., (n - 1)\}$
- This is referred to as the **set of residues**, or **residue classes** (mod *n*). Z_n represents residue classes.
- We can label the residue classes (mod *n*) as [0], [1], [2], ..., [*n* - 1], where $[r] = \{a: a \text{ is an integer}, a \equiv r \pmod{n}\}$
- $Z_4 = \{[0], [1], [2], [3]\}$

The residue classes (mod 4) are

$$[0] = \{\ldots, -16, -12, -8, -4, 0, 4, 8, 12, 16, \ldots\}$$
$$[1] = \{\ldots, -15, -11, -7, -3, 1, 5, 9, 13, 17, \ldots\}$$
$$[2] = \{\ldots, -14, -10, -6, -2, 2, 6, 10, 14, 18, \ldots\}$$
$$[3] = \{\ldots, -13, -9, -5, -1, 3, 7, 11, 15, 19, \ldots\}$$

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 $w + z \equiv 0 \bmod n$

(This table can be found on page 38 in the textbook)

More Number Theory

- This is a start of a new chapter in the text book.
- But was merged here to continue number theory discussion.

Prime Numbers

- Prime numbers only have divisors of 1 and itself (i.e. ±1 and ± p)
  - They cannot be written as a product of other numbers
- Prime numbers are central to number theory
- Prime factorization theorem (or unique factorization theorem or the fundamental theorem of arithmetic):

Any integer $a > 1$ can be factored in a unique way as

$$a = (p1)^{a1} * (p_2)^{a2} * \ldots * (p_t)^{at}$$

where:

- $p1 < p2 < \ldots < p_t$ are prime numbers and
- each a_i is a positive integer

Primes Under 2000

2	101	211	307	401	503	601	701	809	907	1009	1103	1201	1301	1409	1511	1601	1709	1801	1901
3	103	223	311	409	509	607	709	811	911	1013	1109	1213	1303	1423	1523	1607	1721	1811	1907
5	107	227	313	419	521	613	719	821	919	1019	1117	1217	1307	1427	1531	1609	1723	1823	1913
7	109	229	317	421	523	617	727	823	929	1021	1123	1223	1319	1429	1543	1613	1733	1831	1931
11	113	233	331	431	541	619	733	827	937	1031	1129	1229	1321	1433	1549	1619	1741	1847	1933
13	127	239	337	433	547	631	739	829	941	1033	1151	1231	1327	1439	1553	1621	1747	1861	1949
17	131	241	347	439	557	641	743	839	947	1039	1153	1237	1361	1447	1559	1627	1753	1867	1951
19	137	251	349	443	563	643	751	853	953	1049	1163	1249	1367	1451	1567	1637	1759	1871	1973
23	139	257	353	449	569	647	757	857	967	1051	1171	1259	1373	1453	1571	1657	1777	1873	1979
29	149	263	359	457	571	653	761	859	971	1061	1181	1277	1381	1459	1579	1663	1783	1877	1987
31	151	269	367	461	577	659	769	863	977	1063	1187	1279	1399	1471	1583	1667	1787	1879	1993
37	157	271	373	463	587	661	773	877	983	1069	1193	1283		1481	1597	1669	1789	1889	1997
41	163	277	379	467	593	673	787	881	991	1087		1289		1483		1693			1999
43	167	281	383	479	599	677	797	883	997	1091		1291		1487		1697			
47	173	283	389	487		683		887		1093		1297		1489		1699			
53	179	293	397	491		691				1097				1493					
59	181			499										1499					
61	191																		
67	193																		
71	197																		
73	199																		
79																			
83																			
89																			
97																			

(This table can be found on page 44 in the textbook)

Fermat's Theorem

- States the following:

  If p is prime and a is:

  - a positive integer
  - not divisible by p (➔ $a \pmod{p} \neq 0$)

  then $\quad a^{p-1} \equiv 1 \pmod{p}$

  And $\quad a^{p-1} \pmod{p} \equiv 1$

Example:
p=3
a=8

$a^{p-1} \equiv 1 \pmod{p}$
$8^2 = 64 \equiv 1 \pmod{3}$

- An alternate form is:
  - If p is prime and a is a positive integer then

$$a^p \equiv a \pmod{p}$$

$$a^p \pmod{p} \equiv a$$

$a^p \equiv a \pmod{p}$
$8^3 = 512 \equiv 8 \pmod{3}$
$\equiv 2 \pmod{3}$

Euler's Totient Function: ø (n)

- Euler's Totient function ø (n) defined as:
  - the number of positive integers **less than *n*** and **relatively prime to *n*** .
  - By convention: ø(1) = 1
  - If p is a prime number: ø (*p*)= *p*-1
- Example:
  - ø(35) = 24, why?
  - Positive integers less than 35 and relatively prime to 35: 1, 2, 3, 4, 6, 8, 9, 11, 12, 13, 16, 17, 18, 19, 22, 23, 24, 26, 27, 29, 31, 32, 33, 34
  - Hint: 35=5*7. So, 5 (and 5x) and 7 (and 7x) should be excluded from ø(35).

ø (n) and Prime numbers

- Assume:
  - ***p*** *and* ***q*** are prime number
  - $n = p \times q$
- Then:
  - $ø(p) = p-1$
  - $ø(q) = q-1$
  - $ø(n) = ø(p) \times ø(q) = (p-1) \times (q-1)$
  - Note: if $p=q$, then : $ø(n) = (p-1) \times p$
- Example:
  - $ø(35) = ø(5) \times ø(7) = (5-1) \times (7-1) = 24$
  - $ø(25) = (5-1) \times 5 = 20$

Some Values of Euler's Totient Function ø(*n*)

n	$\phi(n)$
1	1
2	1
3	2
4	2
5	4
6	2
7	6
8	4
9	6
10	4

n	$\phi(n)$
11	10
12	4
13	12
14	6
15	8
16	8
17	16
18	6
19	18
20	8

n	$\phi(n)$
21	12
22	10
23	22
24	8
25	20
26	12
27	18
28	12
29	28
30	8

(This table can be found on page 48 in the textbook)

General Formula to Compute ø(n) for any n

If $n = p_1^{e_1} \dots p_k^{e_k}$, where p_i are primes and $e_i > 0$, then

$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_k}\right)$$

- Examples:

$$\phi(900) = 900\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right)\left(1 - \frac{1}{5}\right) = 240$$

$$\phi(25) = 25\left(1 - \frac{1}{5}\right) = 20$$

Euler's Theorem

- States that for every ***a*** and ***n*** that are relatively prime:

$$a^{\text{ø}(n)} \equiv 1 (\text{mod } n)$$

$$a^{\text{ø}(n)} (\text{mod } n) \equiv 1$$

- An alternative form is:

$$a^{\text{ø}(n)+1} \equiv a (\text{mod } n)$$

$$a^{\text{ø}(n)+1} (\text{mod } n) \equiv a$$

Example:

- a=3,   n=10
  ➔ $\phi(10)=4$
- $a^{\text{ø}(n)} = 3^4 = 81 \ (\text{mod } 10) \equiv 1$
- $a^{\text{ø}(n)+1} = 3^5 = 243 \ (\text{mod } 10) \equiv 3$

Fermat's Theorem vs. Euler's Theorem

	Fermat	Euler
Condition	• p is prime • a is a positive integer • $a \pmod p \neq 0$	***a*** and ***n*** that are relatively prime
Formula	$a^{p-1} \pmod p = 1$	$a^{\phi(n)} \pmod n = 1$

Fermat theorem is a special case of **Euler** theorem:

Euler theorem when ***n*** is a prime number = Fermat theorem

- ***n*** is a prime number → $\phi(\boldsymbol{n}) = \boldsymbol{n}-1$
- Euler= $a^{\phi(n)} \pmod n = a^{n-1} \pmod n = 1$
- Fermat: $a^{n-1} \pmod n = 1$

Testing for Primality

- For many cryptographic algorithms, it is necessary to select large prime number.

<u>Example Algorithms:</u>

- **Miller-Rabin Algorithm**
  - It tells if a number is probably a prime.
- **AKS Algorithm**
  - Prior to 2002 there was no known method of efficiently proving the primality of very large numbers. Algorithms produce probabilistic result
  - In 2002 Agrawal, Kayal, and Saxena developed an algorithm that efficiently determines whether a given large number is prime
    - Known as the AKS algorithm
    - Does not appear to be as efficient as the Miller-Rabin algorithm.

We will study this algorithm.
We present two versions:
- Simplified (one iteration): this is the one we will use in our course.
- Full version (multiple iterations)

Miller-Rabin Test: full version

Full Version

Input : n, R
Output:
  "composite" if n is found to be composite
  "probably prime" otherwise

Compute m and k such that: $n-1 = m \times 2^k$
LOOP: repeat R times:
  pick a random integer a in the range [2, n – 2]
  $T \leftarrow a^m \bmod n$
  if T = 1 or T = n – 1 then
    continue **LOOP**
  repeat k – 1 times:
    $T \leftarrow T^2 \bmod n$
    if T = n – 1 then
      continue **LOOP**
  return "composite"
return "probably prime"

This the full version of Miller-Rabin Test.
The main differences between full and simple:
- Full is repeated ***R*** times (simplified runs one iteration)
  - ***R*** determines the accuracy of the test.
  - Larger ***R*** produces more accurate result.
- "probably prime" output is produced when all options are cases are tested.

Simple Version: *R*=1

- Test(**n**) :
  - Inputs : **n**
  - Output : composite, (probably) a prime
- Compute **m** and **k** such that: $n-1 = m \times 2^k$
  - If k=1, n is (probably) prime
- Select $1 < \mathbf{a} < \mathbf{n}$
  - typically a=2 (for easy calculations)
- **Set : $T = a^m$** (mod **n**)

For (j=0 ; j≤ k-1; j++) {
  $T = T^2$ (mod n)
  If (T == 1) return composite
  If (T== -1) return (probably) prime
  // Note: $-1 \equiv (n-1) \pmod{n}$
}

return composite

Miller-Rabin Test (simple)

- Test(**n**) :
  - Inputs : **n**
  - Output : composite, (probably) a prime
- Compute **m** and **k** such that: $n-1 = m \times 2^k$
  - If k=1, n is (probably) prime
- Select $1 < \mathbf{a} < \mathbf{n}$
  - typically a=2 (for easy calculations)
- **Set :** $\mathbf{T} = \mathbf{a}^m \pmod{\mathbf{n}}$

For (j=0 ; j≤ k-1; j++) {

T = T^2 (mod n)

If (T == 1) return composite

If (T== (n-1)) return (probably) prime

// Note: -1 ≡ (n-1) (mod n)

}

return composite

Examples: a=2 (for all test)

n	$n-1=m*2^k$	m	k	Analysis: pick a=2 to evaluate $T= a^m \bmod n$
61	$60=15\times 2^2$	15	2	$T=2^{15} \pmod{61} \equiv 11 \pmod{61}$ j=0: $T^2= (11)^2 \pmod{61} = 60 \pmod{61} = $ **-1** 61 is probably a prime
53	$52=13\times 2^2$	13	2	$T=2^{13} \pmod{53} = 30$ j=0: $T^2= (30)^2 \pmod{53} \equiv 52 \pmod{53} =$ **-1** 53 is probably a prime
27	$26=13\times 2^1$	13	1	**K=1** ➔ **n** is (probably) a prime Wrong prediction! $27=3\times 9$ **(liar)**
29	$28=7\times 2^2$	7	2	$T=2^7 \pmod{29} = 12$ $T^2= (12)^2 \pmod{29} \equiv 28 \pmod{29} =$ **-1** 29 is probably a prime
561	$560=35\times 2^4$	35	4	$T=2^{35} \bmod 561 = 263$ j=0: $T^2= (263)^2 \pmod{561} = 166$ j=1: $T^4= (166)^2 \pmod{561} = 67$ j=2: $T^8= (67)^2 \pmod{561} =$ **1** ➔ **n is composite number** ($561=3\times 11\times 17$)

Examples: a =2, 3 , 5 (and more)

n	n-1=m*2^k	m	k	Analysis
221	$220=55\times2^2$	55	2	a=5 $T=5^{55}$ mod 221 = 112 j=0: $T^2= (112)^2$ mod 221 = 168 j=1: $T^4= (168)^2$ mod 221 = 157 ➔ **n is composite number**
				a=3 $T=3^{55}$ mod 221 = 198 j=0: $T^2= (198)^2$ mod 221 = 87 j=1: $T^4= (87)^2$ mod 221 = 55 ➔ **n is composite number**
				a=2 $T=2^{55}$ mod 221 = 128 j=0: $T^2= (128)^2$ mod 221 = 30 j=1: $T^4= (30)^2$ mod 221 = 16 ➔ **n is composite number**
				a=174 $T=174^{55}$ mod 221 = 47 j=0: $T^2= (47)^2$ mod 221 = 220 ➔ **n is probably a prime (liar)**

That 221 is composite

a=174 is a liar!
In fact, the following are liars:
21, 47, 174, 200

Chinese Remainder Theorem (CRT)

- Believed to have been discovered by the Chinese mathematician Sun-Tsu in around 100 A.D.
- One of the most useful results of number theory
- Says it is possible to reconstruct integers in a certain range from their residues modulo a set of pairwise relatively prime moduli
- Can be stated in several ways

Provides a way to manipulate (potentially very large) numbers mod *M* in terms of tuples of smaller numbers

- This can be useful when *M* is 150 digits or more
- However, it is necessary to know beforehand the factorization of *M*

CRT

- Let:
  - $n_1, n_2, \ldots, n_k$ be pairwise relatively prime integers
  - $M = n_1 \times n_2 \times \ldots \times n_k$
  - $M_i = M/n_i$
- If $a_1, a_2, \ldots, a_k$ are any integers, then there exists **x** modulu M that satisfies system of linear congruencies:

$$\mathbf{x} \equiv a_1 \pmod{n_1}$$
$$\mathbf{x} \equiv a_2 \pmod{n_2}$$
$$\ldots$$
$$\mathbf{x} \equiv a_k \pmod{n_k}$$

Given a system of linear congruencies, CRT solves **X**

where **x** (and ***y's***) are computed as:

$$\mathbf{x} \equiv a_1 M_1 y_1 + a_2 M_2 y_2 + \ldots + a_k M_k y_k \quad \pmod{M}$$
$$M_i y_i \equiv 1 \pmod{n_i}$$

CRT Example

- Solve linear congruencies using CRT (Find x?):
  - $x \equiv 1 \pmod 5$
  - $x \equiv 2 \pmod 6$
  - $x \equiv 3 \pmod 7$
- First: $M = 5 \times 6 \times 7 = 210$
- Second: calculate Mi's and yi's, see table.
- To calculate y_i use:
  - $M_i\, y_i \equiv 1 \pmod{n_i}$
- $\boldsymbol{x} = a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 \mod 210$

  $= 1(42)(3) + 2(35)(5) + 3(30)(4) \mod 210$

  $= 836 \mod 210$

  $= 206 \mod 210$

n_i	a_i	M_i	y_i
$n_1 = 5$	$a_1 = 1$	$M_1 = 6 \times 7 = 42$	$42 \times y_1 \equiv 1 \pmod 5$ $y_1 = 3$
$n_2 = 6$	$a_2 = 2$	$M_2 = 5 \times 7 = 35$	$35 \times y_2 \equiv 1 \pmod 6$ $y_2 = 5$
$n_3 = 7$	$a_3 = 3$	$M_3 = 5 \times 6 = 30$	$30 \times y_3 \equiv 1 \pmod 7$ $y_3 = 4$

CRT Example (2)

- Solve linear congruencies using CRT (Find x?):
  - $x \equiv 1 \pmod 7$
  - $x \equiv 8 \pmod{11}$

- First: $M=7 \times 11 = 77$
- Second: calculate Mi's and yi's, see table.
- To calculate y_i use:
  - $M_i\, y_i \equiv 1 \pmod{n_i}$
- $\mathbf{x} = a_1 M_1 y_1 + a_2 M_2 y_2 \pmod M$

  $= 1\,(11)\,(2) + 8(7)(8) \pmod{77}$

  $= 8 \pmod{77}$

n_i	a_i	M_i	y_i
$n_1 = 7$	$a_1 = 1$	$M_1 = 11$	$11 \times y_1 \equiv 1 \pmod 7$ $y_1 = 2$
$n_2 = 11$	$a_2 = 8$	$M_2 = 7$	$7 \times y_2 \equiv 1 \pmod{11}$ $y_2 = 8$

The powers of an Integer, Modulo **p:** Primitive Roots

- ***a*** is **primitive root** for ***p*** then: $a, a^2, \ldots, a^{p-1}$ are distinct (mod p)
- In the following example, **3** is a primitive root of modulo **7.**
- This is because 3^k mod 7 generates numbers: 1 ..6, as shown below.
- In following slide, 2, 3, 10, 13, 14 and 15 are primitive roots to prime number 19.

$$
\begin{aligned}
3^1 &= 3 = 3^0 \times 3 \equiv 1 \times 3 = 3 \equiv 3 \pmod 7\\
3^2 &= 9 = 3^1 \times 3 \equiv 3 \times 3 = 9 \equiv 2 \pmod 7\\
3^3 &= 27 = 3^2 \times 3 \equiv 2 \times 3 = 6 \equiv 6 \pmod 7\\
3^4 &= 81 = 3^3 \times 3 \equiv 6 \times 3 = 18 \equiv 4 \pmod 7\\
3^5 &= 243 = 3^4 \times 3 \equiv 4 \times 3 = 12 \equiv 5 \pmod 7\\
3^6 &= 729 = 3^5 \times 3 \equiv 5 \times 3 = 15 \equiv 1 \pmod 7\\
3^7 &= 2187 = 3^6 \times 3 \equiv 1 \times 3 = 3 \equiv 3 \pmod 7
\end{aligned}
$$

Powers of Integers, Modulo 19

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^{10}	a^{11}	a^{12}	a^{13}	a^{14}	a^{15}	a^{16}	a^{17}	a^{18}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	4	8	16	13	7	14	9	18	17	15	11	3	6	12	5	10	1
3	9	8	5	15	7	2	6	18	16	10	11	14	4	12	17	13	1
4	16	7	9	17	11	6	5	1	4	16	7	9	17	11	6	5	1
5	6	11	17	9	7	16	4	1	5	6	11	17	9	7	16	4	1
6	17	7	4	5	11	9	16	1	6	17	7	4	5	11	9	16	1
7	11	1	7	11	1	7	11	1	7	11	1	7	11	1	7	11	1
8	7	18	11	12	1	8	7	18	11	12	1	8	7	18	11	12	1
9	5	7	6	16	11	4	17	1	9	5	7	6	16	11	4	17	1
10	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
11	7	1	11	7	1	11	7	1	11	7	1	11	7	1	11	7	1
12	11	18	7	8	1	12	11	18	7	8	1	12	11	18	7	8	1
13	17	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
14	6	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
15	16	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
16	9	11	5	4	7	17	6	1	16	9	11	5	4	7	17	6	1
17	4	11	16	6	7	5	9	1	17	4	11	16	6	7	5	9	1
18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1

(This table can be found on page 57 in the textbook)

Discrete Logarithm

- Let $b \equiv a^i \pmod{p}$ where : $0 \leq i \leq (p\text{-}1)$
  - *i* is referred to as discrete logarithm of the number b for the base *a* (mod *p*).
- We denote: $i = dlog_{a,p}(b)$
- Example: consider $b \equiv 2^i \pmod{19}$ ➔ $i = \text{dlog}_{2,19}(b)$
- $2 \equiv 2^1 \pmod{19}$ ➔ $1 = \text{dlog}_{2,19}(2)$
- $4 \equiv 2^2 \pmod{19}$ ➔ $2 = \text{dlog}_{2,19}(4)$
- $8 \equiv 2^3 \pmod{19}$ ➔ $3 = \text{dlog}_{2,19}(8)$
- $16 \equiv 2^4 \pmod{19}$ ➔ $4 = \text{dlog}_{2,19}(16)$
- $13 \equiv 2^5 \pmod{19}$ ➔ $5 = \text{dlog}_{2,19}(13)$

(a) Discrete logarithms to the base 2, modulo 19

b	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
i= $\log_{2,19}(a)$	18	1	13	2	16	14	6	3	8	17	12	15	5	7	11	4	10	9

Table 2.8

Tables of Discrete Logarithms, Modulo 19

(a) Discrete logarithms to the base 2, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{2,19}(a)$	18	1	13	2	16	14	6	3	8	17	12	15	5	7	11	4	10	9

(b) Discrete logarithms to the base 3, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{3,19}(a)$	18	7	1	14	4	8	6	3	2	11	12	15	17	13	5	10	16	9

(c) Discrete logarithms to the base 10, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{10,19}(a)$	18	17	5	16	2	4	12	15	10	1	6	3	13	11	7	14	8	9

(d) Discrete logarithms to the base 13, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{13,19}(a)$	18	11	17	4	14	10	12	15	16	7	6	3	1	5	13	8	2	9

(e) Discrete logarithms to the base 14, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{14,19}(a)$	18	13	7	8	10	2	6	3	14	5	12	15	11	1	17	16	4	9

(f) Discrete logarithms to the base 15, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\log_{15,19}(a)$	18	5	11	10	8	16	12	15	4	13	6	3	7	17	1	2	14	9

(This table can be found on page 60 in the textbook)

Summary

- Divisibility and the division algorithm
- The Euclidean algorithm
  - Greatest Common Divisor
  - Finding the Greatest Common Divisor
- Modular arithmetic
  - The modulus
  - Properties of congruences
  - Modular arithmetic operations
  - Properties of modular arithmetic
  - Euclidean algorithm revisited
  - The extended Euclidean algorithm
- Prime numbers

- Fermat's Theorem
- Euler's totient function
- Euler's Theorem
- Testing for primality
  - Miller-Rabin algorithm
  - A deterministic primality algorithm
  - Distribution of primes
- The Chinese Remainder Theorem
- Discrete logarithms
  - Powers of an integer, modulo *n*
  - Logarithms for modular arithmetic
  - Calculation of discrete logarithms