More Number Theory

From section 4.3,
with additions

A very useful Theorem

Th. If a and b are positive integers then gcd(a,b) is the smallest positive integer of the form

sa + tb, where s and t are integers.

(note: one of s and t will be positive, the other negative)

Proof. Let
S = {sa + tb : s,t are integers}

a = 1a + 0b, b = 0a + 1b are in S.

Note that the sum of any two elements of S is also in S, and any multiple of an element of S is in S.

Thus, if x,y are in S and we divide y into x, x = qy + r, $0 \le r < y$, then r = x−qy is in S.

Now let d = the smallest positive integer in S = {sa+tb : s,t are integers}

Then for any x in S, d | x, because if x = qd + r, $0 \le r < d$, then r is in S, so r must be 0 by definition of d.

Thus d is a common divisor of a and b.

But every common divisor u of a and b divides every element of S, and hence u divides d. Hence $u \le d$.

Thus d must be gcd(a,b), the greatest common divisor of a and b.

An example

We can use the Euclidean Algorithm and work backwards to get this representation of the gcd.

Let's do gcd(287,91).

1. 287 = 91*3 + 14
2. 91 = 14*6 + 7
3. 14 = 7*2 + 0, so gcd = 7.
4. From 2, 7 = 91 – 14*6
5. From 1, 7 = 91 – (287−91*3)* 6 so
6. 7 = 19*91 − 287*6

Note that this representation of gcd(a,b) as sa + tb isn't unique.

We have

7 = 19*91 −287*6, but also

7 = (19−287)*91+ (−6+91)*287, so

7 = (−268)*91+ (85)*287

For another algorithm, see p 273, 41−45 (6th ed. p. 246, 48−51)

A useful Corollary

Theorem: gcd(a,b) = 1 iff there are integers s and t such that 1 = sa+tb

Proof:

If gcd(a,b) = 1 then 1 = sa+tb by the previous theorem.

Conversely, if sa+tb = 1 for some s,t then 1 must be the smallest positive integer in the set S = {sa+tb : s,t are integers} and hence 1 = gcd(a,b) by the previous theorem.

More useful facts

Lemma (p.271) (p 233, 6th ed). If a | bc and gcd(a,b) = 1 then a | c (a,b,c positive integers).

Proof: if gcd(a,b) = 1 then 1 = sa+tb, so c = sac + tbc. Hence a | c.

Corollary: if p is prime, a_i are integers and p | $a_1a_2..a_n$, then p | a_i for some i.

Proof: for each i, p | a_i or gcd(p, a_i) = 1, and use induction on n.

Fundamental Theorem of Arithmetic

From the Corollary we get the uniqueness part of the Fundamental Theorem of Arithmetic or Unique Factorization Theorem.

Suppose n = $p_1p_2 \ldots p_s = q_1q_2 \ldots q_t$ where the p_i and q_i are distinct primes written in increasing order.

If each $p_i = q_i$ and s=t we are done.

If not, divide out by the common primes and get a smaller n where the p_i, q_j are all distinct (rename the primes, and s,t, for simplicity).

But then p_1 | $q_1q_2 \ldots q_t$ and isn't one of the q_j, a contradiction. So we must have had identical primes in the factorization.

Solving Linear Congruences

Th. (p. 272) If ac ≡ bc (mod m) and gcd(c,m) = 1, then a ≡ b (mod m)

proof: since ac ≡ bc (mod m) we have m | ac−bc = c(a−b).

Since gcd(c,m) = 1, m | a – b, so

a ≡ b (mod m)

Note: this is a cancellation law, like the usual rule ac = bc → a = b if c ≠ 0.

Solving Linear Congruences

Theorem (p.275)(p 234 6th ed): Suppose gcd(a,m) = 1, m > 1. Then an inverse of a modulo m exists and is unique modulo m.

That is, there is an integer s with sa ≡ 1 (mod m) and if ta ≡ 1 (mod m) then s ≡ t (mod m)

proof: Since gcd(a,m) = 1 we have sa + tm = 1 for some integers s,t.

But it follows from this that sa = 1 – tm, so

sa ≡ 1 (mod m)

An example

To find an inverse of 7 modulo 11, we need $s7 + t11 = 1$.

We use trial and error, look at multiples of 7 and 11.

7, 14, 21, 28, …,

11, 22, 33, …. We've found it!

$1 = 22 - 21 = 11*2 + (-3)*7$, so -3 is an inverse to 7.

But we want a positive inverse, so add 11.

$-3 + 11 = 8$. Yup, $7*8 \equiv 1 \pmod{11}$

Following up…

Suppose we want to solve

$7x \equiv 5 \pmod{11}$ for x.

Since we have an inverse to 7, 8,

$8*7*x \equiv 8*5 \pmod{11}$,

$x \equiv 7 \pmod{11}$, since $40 \equiv 7 \pmod{11}$

Check:

$7*7 = 49 \equiv 5 \pmod{11}$.

Chinese Remainder Theorem

Sun-Tsu asked: Is there some x such that

1. $x \equiv 2 \pmod 3$ and
2. $x \equiv 3 \pmod 5$ and
3. $x \equiv 2 \pmod 7$?

For 1, x = 2, 5, 8, 11, 14, 17, 20, 23, 28,…

For 2, x = 3, 8, 13, 18, 23, 28, 33, …

For 3, x = 2, 9, 16, 23, 30, 37,…

So, x = 23 satisfies all three conditions!

And it turns out x is unique mod $3*5*7 = 105$.

Note that if x is a solution, so is $x + n*105$.

The Chinese Remainder Theorem

Th. (p 278) Suppose $m_1, m_2 \ldots m_n$ are pairwise relatively prime positive 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 x modulo $m = m_1 * m_2 \ldots * m_n$

Proof:

Let $M_k = m/m_k = m_1..m_{k-1}m_{k+1}..m_n$

Then $\gcd(M_k, m_k) = 1$ for $k = 1,\ldots,n$.

Hence M_k has an inverse y_k mod m_k, $M_k y_k \equiv 1 \pmod{m_k}$

Let $x = a_1M_1y_1 + a_2M_2y_2 + \ldots + a_nM_ny_n$.

Then $x \equiv a_kM_ky_k \equiv a_k \pmod{m_k}\ \forall k$, since $a_jM_jy_j \equiv 0 \pmod{m_k}$ for $j \neq k$

To see uniqueness, if x and y are two solutions then $x - y \equiv 0 \pmod{m_k}\ \forall k$ and hence $m \mid x - y$, so $x \equiv y \pmod m$.

The example again

$x \equiv 2 \pmod 3$, so $a_1 = 2$, $m_1 = 3$

$x \equiv 3 \pmod 5$, so $a_2 = 3$, $m_2 = 5$

$x \equiv 2 \pmod 7$, so $a_3 = 2$, $m_3 = 7$

Thus $M_1 = 35$, $M_2 = 21$, $M_3 = 15$.

Now $2*35 = 70 \equiv 1 \pmod 3$, let $y_1 = 2$

$21*1 \equiv 1 \pmod 5$, so let $y_2 = 1$

$15*1 \equiv 1 \pmod 7$, so let $y_3 = 1$.

Let $x = 2*35*2 + 3*21*1 + 2*15*1 = 233$.

Now $3*5*7 = 105$, and $233 \equiv 23 \pmod{105}$, so 23 is a solution, unique mod 105.

Chinese Remainder Theorem

The Chinese Remainder Theorem can be used to design systems for doing large number arithmetic.

See page 278 (p. 236, 6^{th} ed.)

Hash Functions

Hash functions are used to maps long keys (e.g. names, id numbers) to array locations. If there are m array locations, a simple method is to convert the key to an integer k and then map to k mod m.

Hashing collisions

A collision is when two keys map to the same array location.

A perfect hash function is designed to produce no collisions.

A collision can be resolved by moving down the array to the next free array location, or by hanging linked lists off the array locations.

Pseudorandom Numbers

It's difficult to generate truly random numbers.

Computers often generate "random" numbers using a linear congruential method. For fixed m, a, c and a seed x_0, we define $x_{n+1} = (ax_n + c) \bmod m$.

Fermat's Little Theorem

Theorem: If p is prime and p does not divide a, then

$a^{p-1} \equiv 1 \pmod p$, and thus also

$a^p \equiv a \pmod p$.

Proof: The numbers a, 2a, 3a, …, (p−1)a are distinct mod p since their pairwise differences are not 0 mod p.

Thus they are 1, 2, 3, …, p−1 in some order, mod p.

So $a*2a*\ldots*(p-1)a \equiv 1*2*\ldots*(p-1) \pmod p$

Dividing both sides by $1*2*3*\ldots*(p-1)$, which is relatively prime to p, we get

$a^{p-1} \equiv 1 \pmod p$, hence $a^p \equiv a \pmod p$

Cryptology

A really simple cryptographic method was used by Julius Caesar. This was to shift each letter right a fixed number of places in the alphabet.

If we encode each letter by its position in the alphabet we can use:
$f(x) = (x + k) \bmod 26$, to shift k places.

Cryptology

The Caesar cypher is very easy to crack, and any cryptographic method which uses a fixed code for each letter is vulnerable to attacks based on the frequency of occurrence of particular letters.

RSA Encryption

RSA encryption exploits the computational difficulty of factoring large numbers to create a public key for encryption and a private key for decryption.

Public: Suitable large integers n and e.

Private: primes p, q, integer d, with $pq = n$, and $de \equiv 1 \bmod (p-1)(q-1)$

RSA Encryption

A block of the message is M, interpreted as a number.

We encrypt it by computing
$C = M^e \bmod n$

Here e is part of the public key. We use an efficient algorithm for computing the power. (Algorithm 5, p. 254)

RSA Decryption

We decrypt using the private key.

e has been selected relatively prime to $(p-1)(q-1)$.

$de \equiv 1 \bmod (p-1(q-1)$, so that
$de = 1 + k(p-1)(q-1)$.

We can arrange for $\gcd(M, pq) = 1$.

Generally M has some random padding for extra security.

Thus, by Fermat's Little Theorem,
$M^{p-1} \equiv 1 \pmod p$, $M^{q-1} \equiv 1 \pmod q$

So $C^d \equiv M*(M^{p-1})^{k(q-1)} \equiv M * 1 \pmod p$,

& $C^d \equiv M*(M^{q-1})^{k(p-1)} \equiv M * 1 \pmod q$.

Since $C^d \equiv M \pmod p$ & $C^d \equiv M \pmod q$

Hence, by the Chinese Remainder Theorem,
$C^d \equiv M \pmod{pq}$ i.e.
$C^d \equiv M \pmod n$ – This is the decryption.

[Since M is a solution and the solution is unique mod n, and C^d is a solution, we have $C^d \equiv M \bmod n$. (Chinese Remainder Theorem)]

Private Key Cryptography

The RSA algorithm uses a fair bit of computation, so in practice it is used not for exchanging large messages, but for a secure exchange of private keys which can then be used to exchange large messages efficiently and securely using DES or AES, symmetric key algorithms, whose computational cost is cheap. See Wikipedia for more info.

Matrices

A **matrix** is a rectangular array of numbers.
A matrix with m rows and n columns is called an m×n matrix.

Example: $A = \begin{bmatrix} -1 & 1 \\ 2.5 & -0.3 \\ 8 & 0 \end{bmatrix}$ is a 3×2 matrix.

A matrix with the same number of rows and columns is called **square**.

Two matrices are **equal** if they have the same number of rows and columns and the corresponding entries in every position are equal.

Matrices

A general description of an m×n matrix A = [a_{ij}]:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \qquad \begin{bmatrix} a_{1j} \\ a_{2j} \\ \cdot \\ \cdot \\ \cdot \\ a_{mj} \end{bmatrix}$$

j-th column of A

$$[a_{i1}, a_{i2}, \dots, a_{in}]$$

i-th row of A

Matrix Addition

Let A = [a_{ij}] and B = [b_{ij}] be m×n matrices.
The sum of A and B, denoted by A+B, is the m×n matrix that has $a_{ij} + b_{ij}$ as its (i, j)th element.
In other words, A+B = [$a_{ij} + b_{ij}$].

Example:

$$\begin{bmatrix} -2 & 1 \\ 4 & 8 \\ -3 & 0 \end{bmatrix} + \begin{bmatrix} 5 & 9 \\ -3 & 6 \\ -4 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 10 \\ 1 & 14 \\ -7 & 1 \end{bmatrix}$$

Matrix Multiplication

Let A be an m×k matrix and B be a k×n matrix.
The **product** of A and B, denoted by AB, is the m×n matrix with (i, j)th entry equal to the sum of the products of the corresponding elements from the i-th row of A and the j-th column of B.

In other words, if AB = [c_{ij}], then

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \ldots + a_{ik}b_{kj} = \sum_{t=1}^{k} a_{it}b_{tj}$$

Matrix Multiplication

A more intuitive description of calculating C = AB:

$$A = \begin{bmatrix} 3 & 0 & 1 \\ -2 & -1 & 4 \\ 0 & 0 & 5 \\ -1 & 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 2 & 1 \\ 0 & -1 \\ 3 & 4 \end{bmatrix}$$

- Take the first column of B
- Turn it counterclockwise by 90° and superimpose it on the first row of A
- Multiply corresponding entries in A and B and add the products: 3·2 + 0·0 + 1·3 = 9
- Enter the result in the upper-left corner of C

Matrix Multiplication

- Now superimpose the first column of B on the second, third, …, m-th row of A to obtain the entries in the first column of C (same order).
- Then repeat this procedure with the second, third, …, n-th column of B, to obtain to obtain the remaining columns in C (same order).
- After completing this algorithm, the new matrix C contains the product AB.

Matrix Multiplication

Let us calculate the complete matrix C:

$$A = \begin{bmatrix} 3 & 0 & 1 \\ -2 & -1 & 4 \\ 0 & 0 & 5 \\ -1 & 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 2 & 1 \\ 0 & -1 \\ 3 & 4 \end{bmatrix}$$

$$C = \begin{bmatrix} 9 & 7 \\ 8 & 15 \\ 15 & 20 \\ -2 & -2 \end{bmatrix}$$

Identity Matrices

The **identity matrix of order n** is the n×n matrix $I_n = [\delta_{ij}]$, where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$:

$$A = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ 0 & 0 & \ldots & 1 \end{bmatrix}$$

Multiplying an m×n matrix A by an identity matrix of appropriate size does not change this matrix:

$AI_n = I_mA = A$

Powers and Transposes of Matrices

The **power function** can be defined for **square** matrices. If A is an n×n matrix, we have:

$A^0 = I_n$,
$A^r = AAA\ldots A$ (r times the matrix A)

The **transpose** of an m×n matrix $A = [a_{ij}]$, denoted by A^t, is the n×m matrix obtained by interchanging the rows and columns of A.

In other words, if $A^t = [b_{ij}]$, then $b_{ij} = a_{ji}$ for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$.

Powers and Transposes of Matrices

Example: $A = \begin{bmatrix} 2 & 1 \\ 0 & -1 \\ 3 & 4 \end{bmatrix} \quad A^t = \begin{bmatrix} 2 & 0 & 3 \\ 1 & -1 & 4 \end{bmatrix}$

A square matrix A is called **symmetric** if $A = A^t$.
Thus $A = [a_{ij}]$ is symmetric if $a_{ij} = a_{ji}$ for all $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$.

$$A = \begin{bmatrix} 5 & 1 & 3 \\ 1 & 2 & -9 \\ 3 & -9 & 4 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 3 & 1 \\ 1 & 3 & 1 \\ 1 & 3 & 1 \end{bmatrix}$$

A is symmetric, B is not.