Discrete Math for Computer Science

Peter Schaefer

Freshman - Kutztown

Contents

1 Integer Properties

1.1 The Division Algorithm

In **integer division**, the input and output values must always be integers. For example, when 9 is divided by 4, the answer is 2 with a remainder of 1, instead of 2.25.

Divides

Let x and y be two integers. Then x *divides* y, $x \mid y$, if and only if $x \neq 0$ and there is an integer k such that $y = kx$. If there is no such integer or if $x = 0$, then x does not divide y, $x \nmid y$. If $x \mid y$, then y is said to be a *multiple* of x, and x is a *factor* or *divisor* of y.

Theorem: Divisibility and linear combinations

Let $x, y,$ and z be integers. If $x \mid y$ and $x \mid z$, then $x \mid (sy + tz)$ for any integers s and t.

Proof. Since $x \mid y$, then $y = kx$ for some integer k. Similarly, since $x \mid z$, then $z = jx$ for some integer j. A linear combination of y and z can be expressed as:

$$sy + tz = s(kx) + t(jx) = (sk + tj)x.$$

For some integers s and t. Since $sy + tz$ is an integer multiple of x, then $x \mid (sy + tz)$. □

Quotients and remainders

If $x \nmid y$, then there is s non-zero remainder when x is divided into y. The **Division Algorithm**, states that the result of the division and the remainder are unique.

Theorem: The Division Algorithm

Let n be an integer and let d be a positive integer. Then, there are unique integers q and r, with $0 \leq r \leq d-1$, such that $n = qd + r$.

Integer division definitions

In the Division Algorithm, q is called the **quotient** and r is called the **remainder**. The operations **div** and **mod** produce the quotient and the remainder as a function of n and d.

$$q = n \operatorname{div} d$$
$$r = n \bmod d$$

Here is some examples of computing div and mod:

$$15 \bmod 6 = 3 \qquad -11 \bmod 4 = 1$$
$$15 - 2 \cdot 6 = 3 \qquad -11 - (-3) \cdot 4 = 1$$

$$15 \operatorname{div} 6 = 2 \qquad -11 \operatorname{div} 4 = -3$$
$$\frac{15 - 3}{2} = 2 \qquad \frac{-11 - 1}{4} = -3$$

1.2 Modular arithmetic

Given a finite set of integers, we can define addition and multiplication on the elements in the set such that after every operation, we apply a modular function equal to the cardinality of the set.

- **addition mod** m

  the operation defined by adding two numbers and applying $\bmod m$ to the result

- **multiplication mod** m

  the operation defined by multiplying two numbers and applying $\bmod m$ to the result

The set $\{0, 1, 2, \ldots, m-1\}$ along with addition and multiplication mod m defines a closed mathematical system with m elements called a **ring**. The ring $\{0, 1, 2, \ldots, m-1\}$ with addition and multiplication mod m is denoted by $\mathbb{Z}_m$.

Applications

A common way to organize data is to maintain an array called a **hash table** which is slightly larger than the number of data items to be stored. A bldhash function is used to map each data item to a location in the array. Modulus is used to keep the results from a hash function in the range of the hash table.

Computers use functions called bldpseudo-random number generators that produce numbers having many of the statistical properties of random numbers but are in fact deterministically generated. Modulus is used to keep these pseudo-random number generators in a certain range when used.

Congruence mod m

Let $m \in \mathbb{Z} > 1$. Let x and y be any two integers. Then bld$x$ is congruent to $y \bmod m$ if $x \bmod m = y \bmod m$. The fact that x is congruent to $y \bmod m$ is denoted

$$x \equiv y \pmod{m}.$$

Theorem: Alternate characterization of congruence mod m

Let $m \in \mathbb{Z} > 1$. Let x and y be any two integers. Then $x \equiv y \pmod{m}$ if and only if $m \mid (x - y)$.

Proof. First suppose that $x \equiv y \pmod{m}$. By definition $xtmodm = y \bmod m$. We define the variable r to be the value of $x \bmod m = y \bmod m$. Therefore, $x = r + km$ for some integer k and $y = r + jm$ for some integer j. Then

$$x - y = (r + km) - (r + jm) = (k - j)m.$$

Since $(k - j)$ is an integer, $m \mid (x - y)$.
Now suppose that $m \mid (x - y)$. Then $(x - y) = tm$ for some integer t. Let r be the value of $x \bmod m$. Then $x = r + km$ for some integer k. The integer y can be expressed as

$$y = x - (x - y) = (r + km) - tm = r + (k - t)m.$$

Since r is an integer in the range from 0 to $m - 1$, r is the unique remainder when y is divided by m. Therefore $r = y \bmod m = x \bmod m$, and by definition $x \equiv y \pmod{m}$. □

Precedence of the mod operation

$$6 + 2 \bmod y = 6 + (2 \bmod 7) = 8$$
$$6 \cdot 2 \bmod 7 = (6 \cdot 2) \bmod 7 = 5$$

However, in general it is best to just use parentheses in order to clarify which operations should be performed first.

Theorem: Computing arithmetic operations mod m

Let m be an integer larger than 1. Let x and y be any integers. Then

$$[(x \bmod m) + (y \bmod m)] \bmod m = [x + y] \bmod m$$
$$[(x \bmod m)(y \bmod m)] \bmod m = [x + y] \bmod m$$

1.3 Prime factorizations

A number p is **prime** if it is an integer greater than 1 and its only factors are 1 and p. A positive integer is **composite** if it has a factor other than 1 or itself. Every integer greater than 1 is either prime of composite. Every positive integer greater than one can be expressed as a product of primes called its **prime factorization**. Moreover, the prime factorization is unique up to ordering of the factors.

Theorem: The Fundamental Theorem of Arithmetic

Every positive integer other than 1 can be expressed uniquely as a product of prime numbers where the primes factors are written in non-decreasing order.

Examples of prime factorizations in non-decreasing order

$$112 = 2^4 \cdot 7$$
$$612 = 2^2 \cdot 3^3 \cdot 17$$
$$243 = 3^5$$
$$17 = 17$$

Greater common divisors and least common multiples

- The **greatest common divisor (gcd)** of integers x and y that are not both zero is the largest integer that is a factor of both x and y.
- The **least common multiples (lcm)** of non-zero integers x and y is the smallest positive integer that is an integer multiple of both x and y.

Two numbers are **relatively prime** if their greatest common divisor is 1.

Theorem: GCD and LCM from prime factorizations

Let x and y be two positive integers with prime factorizations expressed using a common set of primes as:

$$x = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdots p_r^{\alpha_r}$$
$$y = p_1^{\beta_1} \cdot p_2^{\beta_2} \cdots p_r^{\beta_r}$$

The p_i's are all distinct prime numbers. The exponents α_i's and β_i's are non-negative integers. Then:

- $x \mid y$ if and only if $\alpha_i \leq \beta_i$ for all $1 \leq i \leq r$
- $\gcd(x, y) = p_1^{\min(\alpha_1, \beta_1)} \cdot p_2^{\min(\alpha_2, \beta_2)} \cdots p_r^{\min(\alpha_r, \beta_r)}$
- $\text{lcm}(x, y) = p_1^{\max(\alpha_1, \beta_1)} \cdot p_2^{\max(\alpha_2, \beta_2)} \cdots p_r^{\max(\alpha_r, \beta_r)}$

1.4 Factoring and primality testing

A **brute force algorithm** solves a problem by exhaustively searching all positive solutions without using an understanding of the mathematical structure in the problem to eliminate steps.

Theorem: Small Factors

If N is a composite number, then N has a factor greater than 1 and at most $\sqrt{N}$

Theorem: Infinite number of primes

There are an infinite number of primes.

Proof. Suppose that there are a finite number of primes. Since there are only a finite number, they can be listed:

$$p_1, p_2, \ldots, p_k$$

Take the product of all the primes and add 1. Call the resulting number N:

$$N = (p_1 \cdot p_2 \cdots p_k) + 1$$

The number N is larger than all of the primes numbers that were listed, so it must not be prime. Since N is a composite number, it is the product of at least two primes by the Fundamental Theorem of Arithmetic. There N is divisible by some prime p_j. Let

$$\frac{N}{p_j} = \frac{(p_1 \cdot p_2 \cdots p_k)}{p_j} + \frac{1}{p_j}$$

Note that p_j is one of the prime factors in $(p_1 \cdot p_2 \cdots p_k)$, so $(p_1 \cdot p_2 \cdots p_k)/p_j$ is an integer. However, $1/p_j$ is not an integer. Since N/p_j is the sum of two terms, one of which is an integer and the other of which is not an integer, then N/p_j is not an integer. This contradicts the fact that p_j evenly divides N. □

The Prime Number Theorem

Let $\pi(x)$ be the number of prime numbers in the range from 2 through x. Then

$$\lim_{x \to \infty} \frac{\pi(x)}{x/\ln x} = 1.$$

Another way to interpret the Prime Number Theorem is that if a random number is selected from the range 2 to x, then the likelihood that the selected number is prime is roughly $1/\ln x$.

1.5 Greatest common factor divisor and Euclid's algorithm

There is an efficient way to compute the gcd of two numbers without finding their prime factorizations. The algorithm presented in this subsection is in wide use today and is attributed to the Greek mathematician Euclid who lived around 300 B.C. The basis of the algorithm is the following theorem:

GCD Theorem

Let x and y be two positive integers. Then $\gcd(x, y) = \gcd(y \bmod x, x)$.

Euclid's Algorithm for finding the greatest common divisor

```
Input: Two positive integers, x and y.
Output: gcd(x, y)

If(y < x)
  Swap x and y.
r := y mod x

While(r != 0)
  y := x
```

```
   x := r
   r := y mod x
End-While

Return(x)
```

Sample execution of Euclid's algorithm for $\gcd(675, 210)$:

675	210	$675 \bmod 210$		
	210	45	$210 \bmod 45$	
		45	30	$45 \bmod 30$
			30	$30 \bmod 15$
				15 0

The last non-zero number was 15, so $\gcd(675, 210) = 15$.

Expressing $\gcd(x, y)$ as a linear combination of x and y

Let x and y be integers, then there are integers s and t such that

$$\gcd(x, y) = sx + ty.$$

The values for s and t in the theorem above can be found by a series of substitutions using the equation from each iteration. The algorithm used to find the coefficient, s and t, such that $\gcd(x, y) = sx + ty$, is called the **Extended Euclidean Algorithm**.

The Extended Euclidean Algorithm

		y	x	r
675	210	45	30	15
				$r = y \bmod x$
				$r = y - (y \operatorname{div} x) \cdot x$
				$15 = 45 - (45 \operatorname{div} 30) \cdot 30$
				$\mathbf{15 = 45 - 1 \cdot 30}$
			30	$= 210 - (210 \operatorname{div} 45) \cdot 45$
			30	$\mathbf{= 210 - 4 \cdot 45}$
		45	=	$675 - (675 \operatorname{div} 210) \cdot 210$
		45	=	$\mathbf{675 - 3 \cdot 210}$

We can use the bolded equations to solve for $15 = c \cdot 210 + d \cdot 675$.

$$\begin{aligned} 15 &= 45 - 30 \\ &= 45 - (210 - 4 \cdot 45) \\ &= 5 \cdot 45 - 210 \\ &= 5 \cdot (675 - 3 \cdot 210) - 210 \\ &= 5 \cdot 675 - 16 \cdot 210 \end{aligned}$$

Now we have the full answer and expansion for $\gcd(675, 210)$.

$$\gcd(675, 210) = 15 = 5 \cdot 675 - 16 \cdot 210.$$

The Multiplicative Inverse mod n

A **multiplicative inverse mod** n, or just **inverse mod** n, of an integer x, is an integer $s \in \{1, 2, \ldots, n-1\}$ such that $sx \bmod n = 1$.

For example, 3 is an inverse of 7 mod 10 because $3 \cdot 7 \bmod 10 = 1$. The number 7 is an inverse of 5 mod 17 because $7 \cdot 5 \bmod 17 = 1$. It is possible for a number to be its own multiplicative inverse mod n. For example, 7 is the inverse of 7 mod 8 because $7 \cdot 7 \bmod 8 = 1$.

Not every number has an inverse mod n. For example, 4 does not have an inverse mod 6. The condition is that x has an inverse mod n if and only if x and n are relatively prime.

The Extended Euclidean Algorithm can be used to find the multiplicative inverse of $x \bmod n$ when is exists.

- If $\gcd(x, n) \neq 1$, then x does not have a multiplicative inverse mod n.
- If x and n are relatively prime, then the Extended Euclidean Algorithm finds integers s and t such that $1 = sx + tn$.
- $sx - 1 = -tn$. Therefore, $(sx \bmod n) = (1 \bmod n)$. If $A - B$ is a multiple of n then $(A \bmod n) = (B \bmod n)$.
- $(s \bmod n)$ is the unique multiplicative inverse of x in $\{0, 1, \ldots, n-1\}$.

For example, suppose that Euclid's Algorithm returns

$$\gcd(31, 43) = 1 = 13 \cdot 34 - 18 \cdot 31$$

The coefficient of 31 is -18. Therefore, the multiplicative inverse of $31 \bmod 43$ is $(-18 \bmod 43) = 25$.

1.6 Number representation

A digit in binary is called a **bit**. In binary notation, each place value is a power of 2. Numbers represented in **base** b require b distinct symbols and each place value is a power of b.

Theorem: Number representation

For an integer $b > 1$. Every positive integer n can be expressed uniquely as

$$n = a_k \cdot b^k + a_{k-1} \cdot b^{k-1} + \cdots + a_1 \cdot b^1 + a_0 \cdot b^0,$$

where k is a non-negative integer, and each a_i is an integer in the range from 0 to $b-1$, and $a_k \neq 0$. The representation of n base b is called the **base b expansion of** n and is denoted by $(a_k a_{k-1} \ldots a_1 a_0)_b$.

Hexadecimal Numbers

In **hexadecimal** notation (or **hex** for short), numbers are represented in base 16. Typically, the set of symbol, in order of value, is

$$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}.$$

Additionally, here are the first 15 hexadecimal digits and correspond encodings in decimal and binary.

Hex	0	1	2	3	4	5	6	7
Decimal	0	1	2	3	4	5	6	7
Binary	0	1	10	11	100	101	110	111

Hex	8	9	A	B	C	D	E	F
Decimal	8	9	10	11	12	13	14	15
Binary	1000	1001	1010	1011	1100	1101	1110	1111

Since both hexadecimal and binary are powers of 2, there is an easy way to translate between the binary expansion and the hexadecimal expansion of a number. Groups of 4 binary digits can be directly translated into hexadecimal digits. Here is an example

1	1101	0101	1110	1000
1	D	5	E	8

$$1, 1101, 0101, 1110, 1000_2 = 1D5E8_{16}$$

Hexadecimal notation is particularly useful in computer science because each hexadecimal digit can be used to represent a 4-bit binary number. A byte, which consists of 8 bits, can be represented by a 2-digit hexadecimal number. Two hexadecimal digits is easier for a human to recognize and remember than 8 bits.

Converting decimal numbers to base b

$$[\text{Base } b \text{ expansion of } (n \operatorname{div} b)]\,[n \bmod b]$$

Here is an example of 1161_{10} converted to base 7:

		$\ulcorner$	1161
		$1161 \operatorname{div} 7$	$1161 \bmod 7$
	$\ulcorner$	165	6
	$165 \operatorname{div} 7$	$165 \bmod 7$	|
$\ulcorner$	23	4	|
$23 \operatorname{div} 7$	$23 \bmod 7$	|	|
3	2	|	|
3	2	4	6

The number of digits required to represent a number

$$n = (\underbrace{[b-1][b-1]\ldots[b-1]}_{k\text{digits}})_b = (1\underbrace{00\ldots0}_{k0's})_b - 1 = b^k - 1$$

The number of digits required to expressed the base b expansion of a positive integer n is $\lceil \log_b(n+1) \rceil$. For example, the number of digits required to express the base 2 expansion of 13 is:

$$\lceil \log_2(13+1) \rceil = \lceil \log_2(14) \rceil = \lceil 3.8 \rceil = 4.$$

1.7 Fast exponentiation

Computing a power of x by repeated multiplication by x

$$p := x \xrightarrow{p:=p*x} p = x^2 \xrightarrow{p:=p*x} p = x^3 \xrightarrow{p:=p*x} p = x^4 \cdots$$

Computing a power of x by repeated squaring

$$p := x \xrightarrow{p:=p*p} p = x^2 \xrightarrow{p:=p*p} p = x^4 \xrightarrow{p:=p*p} p = x^8 \cdots$$

In order to compute x^y, examine the binary expansion of y:

$$y = a_k \cdot 2^k + a_{k-1} \cdot 2^{k-1} + \cdots + a_1 \cdot 2^1 + a_0 \cdot 2^0$$

We can use this expanded form of y to represent x as a sum of perfect squares

$$x^y = x^{a_k \cdot 2^k + a_{k-1} \cdot 2^{k-1} + \cdots + a_1 \cdot 2^1 + a_0 \cdot 2^0} = x^{a_k \cdot 2^k} \cdot x^{a_{k-1} \cdot 2^{k-1}} \cdots x^{a_1 \cdot 2^1} \cdot x^{a_0 \cdot 2^0}$$

Each coefficient a_j is either 0 or 1. If $a_j = 0$ for some j, then

$$x^{a_j \cdot 2^j} = x^0 = 1$$

and the corresponding factor can be omitted from the product. Here is an example, computing 7^{11}.

$$\begin{aligned} 7^{11} &= 7^{[(1\cdot 2^3)+(0\cdot 2^2)+(1\cdot 2^1)+(1\cdot 2^0)]} \\ &= 7^{(1\cdot 8)} \cdot 7^{(0\cdot 4)} \cdot 7^{(1\cdot 2)} \cdot 7^{(1\cdot 7)} \\ &= 7^8 \cdot 1 \cdot 7^2 \cdot 7^1 \\ 7^1 1 &= 5,764,801 \cdot 49 \cdot 7 \\ &= 1,977,326,743 \end{aligned}$$

An iterative algorithm for fast exponentiation

```
Input: Positive integers x and y.
Output: x^y

p := 1  // p holds the partial result
s := x  // s holds the current x^{2^j}
r := y  // r is used to compute the binary expansion of y

while(r > 0)
  if(r mod 2 = 1)
    p := p * s
  s := s * s
  r := r div 2
End-While

Return(p)
```

An iterative algorithm for fast modular exponentiation

```
Input: Positive integers x, y and n.
Output: x^y mod n

p := 1  // p holds the partial result
s := x  // s holds the current x^{2^j}
r := y  // r is used to compute the binary expansion of y

while(r > 0)
  if(r mod 2 = 1)
    p := p * s mod n
  s := s * s mod n
  r := r div 2
End-While

Return(p)
```

1.8 Introduction to cryptography

Cryptography is the science of protecting and authenticating data and communication. Cryptography is ubiquitous in the electronic age in which sensitive information such as credit card numbers and passwords are sent over the internet on a daily basis.

A cryptosystem is a system by which a **sender** sends a message to a **receiver**. The sender **encrypts** the message so that if an eavesdropper learns the transmitted message, they will be unable to recover the original message. The unencrypted message is called **plaintext** and the encrypted message is called the **ciphertext**. The receiver must have a **secret key** that allows him to **decrypt** the ciphertext to obtain the original plaintext.

Sending an Encrypted Text Message via a Secret Key

1. Alice wants to send the message "MEET AT DAWN" to Bob. Alice converts the text message to the number 13050520270129270401231 4.

2. Alice encrypts the numerical message with her copy of the secret key.

3. Alice sends encrypted message to Bob. Eve cannot read the encrypted message without the secret key.
4. Bob decrypts the encrypted message using the secret key to get 130505202701292704012314 which he then converts to "MEET AT DAWN".

Encryption and Decryption Functions

Consider a simple cryptosystem in which the set of all possible plaintexts come from $\mathbb{Z}_N$ for some integer N. Alice and Bob share a secret number $k \in \mathbb{Z}_N$. The security of their encryption scheme rests on the assumption that no one besides them knows the number k. To encrypt a plaintext $m \in \mathbb{Z}_n$, Alice computes:

$$c = (m + k) \bmod N \qquad \text{(encryption)}$$

Alice sends the ciphertext c to Bob. When Bob receives the ciphertext c, he decrypts c as follows:

$$m = (c - k) \bmod N \qquad \text{(decryption)}$$

The simple encryption scheme presented here is an example of symmetric key cryptography. In a **symmetric key cryptosystem**, Alice and Bob must meet in advance to decide on the value of a shared secret key.

Simple encryption scheme requirements

- If $m \neq m'$ and $m,\ m' \in \mathbb{Z}_N$ then $(m + k) \bmod N \neq (m' + k) \bmod N$ (no two distinct plaintexts map to same ciphertext).
- If $m \in \mathbb{Z}_N$ then $(((m+k) \bmod N) - k) \bmod N = m$ (decryption scheme is inverse of encryption scheme).

However, this encryption method is not terribly secure, and is not used in real world conditions because better methods exist.

1.9 The RSA cryptosystem

In **public key cryptography**, Bob has an **encryption key** that he provides *publicly* so that anyone can use it to send him an encrypted message. Bob holds a matching **decryption key** that he keeps *privately* to decrypt messages. While anyone can use the public key to encrypt a message, the security of the scheme depends on the fact that it is difficult to decrypt the message without having the matching private decryption key.

1. Alice encrypts her message to Bob using the public key.
2. Alice sends the encrypted message to Bob. Eve cannot read the encrypted message.
3. Bob decrypts the message using his private key.

Preparation of public and private keys in RSA

1. Bob selects two large prime numbers, p and q.
2. Bob computes $N = pq$ and $\phi = (p - 1)(q - 1)$.
3. Bob finds an integer e such that $\gcd(e, \phi) = 1$.
4. Bob computes the multiplication inverse of $e \bmod \phi$: an integer d such that $(ed \bmod \phi) = 1$.
5. Public (encryption) key: N and e.
6. Private (decryption) key: d.

The RSA scheme requires that m, the message, is an integer in $\mathbb{Z}_N$ and is not a multiple of p or q. Since p and q are primes with hundreds of digits, it is extremely unlikely that m is a multiple of primes p or q. Alice encrypts her plaintext using e and N to produce ciphertext c as follows:

$$c = m^e \bmod N \qquad \text{(encryption)}$$

Alice transmits c to Bob. Bob decrypts the ciphertext using d to recover m from c:

$$m = c^d \bmod N \qquad \text{(decryption)}$$

Number theoretic fact to establish correctness of RSA

Let p and q be prime numbers and $pq = N$. Suppose that $m \in \mathbb{Z}_n$ and $\gcd(m, N) = 1$. Then $m^{(p-1)(q-1)} \bmod N = 1$.

2 Introduction to Counting

2.1 Sum and Product Rules

The two most basic rules of counting are the sum and product rule. The **product rule** provides a way to count sequences.

Theorem: The Product Rule

Let $A_1, A_2, \ldots, A_n$ be finite sets. Then,

$$|A_1 \times A_2 \times \cdots \times A_n| = |A_1| \cdot |A_2| \cdots |A_n|$$

Counting Strings

If Σ is a set of characters (called an **alphabet**) then Σ^n is the set of all string of length n whose characters come from the set Σ. The product rule can be applied directly to determine the number of strings of a given length over a finite alphabet:

$$|\Sigma^n| = |\underbrace{\Sigma \times \Sigma \times \cdots \Sigma}_{n \text{ times}}| = \underbrace{|\Sigma| \cdot |\Sigma| \cdots |\Sigma|}_{n \text{ times}} = |\Sigma|^n$$

Theorem: The Sum Rule

Consider n sets, $A_1, A_2, \ldots, A_n$. If the sets are pairwise disjoint, meaning that $A_i \cap A_j = \emptyset$ for $i \neq i$, then

$$|A_1 \cup A_2 \cup \cdots \cup A_n| = |A_1| + |A_2| + \cdots + |A_n|$$

2.2 The Bijection Rules

One way to approach difficult counting problems is to show that the cardinality of the set to be counted is equal to the cardinality of a set that is easy to count. The **bijection rule** says that if there is a bijection from one set to another then the two sets have the same cardinality.

A function f from a set S to a set T is called a bijection if and only if f has a well-defined *inverse*, f^{-1}.

The Bijection Rule

Let S and T be two finite sets. If there is a bijection from S to T, then $|S| = |T|$

The k-to-1 Rule

Let X and Y be finite sets. The function $f : X \to Y$ is a **k-to-1 correspondence** if for every $y \in Y$, there are exactly k difference $x \in x$ such that $f(x) = y$.

Suppose there is a k-to-1 correspondence from a finite set A to a finite set B. Then

$$|B| = \frac{|A|}{k}.$$

2.3 The generalized product rule

The **generalized product rule** says that in selecting an item from a set, if the number of choices at each step does not depend on previous choices made, then the number of items in the set is a product of the number of choices in each step.

Generalized Product Rule

Consider a set S of sequences of k items. Suppose there are:

- n_1 choices for the first item.
- For every possible choice for the first item, there are n_2 choice for the second item.
- For every possible choice for the first and second items, there are n_3 choices for the third item.
- $\vdots$
- For every possible choice for the first $k-1$ items, there are n_k choices for the k-th item.

Then $|S| = n_1 \cdot n_2 \cdots n_k$.

2.4 Counting permutations

An **r-permutation** is a sequence of r items with **no repetitions**, all taken from the same set. In a sequence, order matters, so (a, b, c) is different from (b, a, c).

The number of r-permutations from a set with n elements

Let r and n be positive integers with $r \leq n$. The number of r-permutations from a set with n elements is denoted by $P(n, r)$:

$$P(n, r) = \frac{n!}{(n-r)!} = n(n-1)\cdots(n-r+1)$$

2.5 Counting subsets

A subset of size r is called an **r-subset**. An r-subset is sometimes referred to as an **r-combination**. In a subset, order does not matter, so $\{a, b, c\}$ is the same as $\{b, a, c\}$. The counting rules for sequences and subsets are commonly referred to as "*permutations* and *combinations*". The term "combination" is the context of counting is another word for "subset".

Counting Subsets: 'n choose r' notation

The number of ways to select an r-subset from a set of size n is:

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}.$$

$\binom{n}{r}$ is read as "n choose r". The notation $C(n, r)$ is sometimes used for $\binom{n}{r}$.

We can calculated an expression for $\binom{n}{n-r}$ by replacing r with $n-r$ in the expression for $\binom{n}{r}$.

$$\binom{n}{n-r} = \frac{n!}{(n-r)!(n-(n-r))!} = \frac{n!}{(n-r!)r!} = \binom{n}{r}$$

This is an **identity** for r-subsets.

2.6 Subset and permutation examples

Two different cat selection problems: Subset vs. Permutations

Consider two closely related counting problems:

1. A family goes to the animal shelter to adopt 3 cats. The shelter has 20 different cats from which to select. How many ways are there for the family to their selection?
2. Three different families go to the animal shelter to adopt a cat. Each family will select one cat. How many ways are there for the families to make their selection? (Note that which family gets which cat matters).

In the first problem, the number is ways to make the selection is $\binom{20}{3}$ because the order in which the cats are selected is not important. The outcome is a 3-subset.

In the second problem, the specific cat selected by each family is important. Additionally, no cat can belong to two families. Thus, the answer is $P(20, 3) = 20 \cdot 19 \cdot 18$. The outcome is a 3-permutation.

2.7 Counting by complement

Counting by complement is a technique for counting the number of elements in a set S that have a property by counting the total number of elements in S and subtracting the number of elements in S that do not have the property.

$$|P| = |S| - |\bar{P}|$$

Suppose we want to count the number of people in a room with red hair. We know that there are 20 people in the room and exactly 12 of them do not have red hair. Then we can deduce that the number of people in the room with red hair is 20 - 12 = 8.

2.8 Permutations with repetitions

A **permutation with repetition** is an ordering of a set of items in which some of the items may be identical to each other. To illustrate with a smaller example, there are $3! = 6$ permutations of the letters CAT, because the letters in CAT are all different. However, there are only 3 different ways to scramble the letters in DAD: ADD, DAD, DDA.

Formula for Counting Permutations with Repetition

The number of distinct sequences with $n_1 1's, n_2's, \ldots, n_k k's$, where $n = n_1 + n_2 + \cdots + n_k$ is

$$\frac{n!}{n_1! n_2! \cdots n_k!}$$

The formula for permutations with repetition is derived from repeated use of the formula for counting r-subsets:

$$\begin{aligned}
&\binom{n}{n_1}\binom{n-n_1}{n_2}\binom{n-n_1-n_2}{n_3}\cdots\binom{n-n_1-n_2-\cdots-n_{k-1}}{n_k} \\
=&\frac{n!}{n_1!(n-n_1)!}\cdot\frac{(n-n_1)!}{n_2!(n-n_1-n_2)!}\cdot\frac{(n-n_1-n_2)!}{n_3!(n-n_1-n_2-n_3)!}\cdots\frac{(n-n_1-n_2-\cdots-n_{k-1})!}{n_k!0!} \\
=&\frac{n!}{n_1! n_2! \cdots n_k!}
\end{aligned}$$

2.9 Counting multisets

A set is a collection of distinct items. A **multiset** is a collection that can have multiple instances of the same kind of item. When $\{1, 2, 2, 3\}$ is viewed as a set, the repetitions don't matter and $\{1, 2, 2, 3\} = \{1, 2, 3\}$. However, when $\{1, 2, 2, 3\}$ is viewed as a multiset, then the fact there are two occurrences of 2 is important, and $\{1, 2, 2, 3\} \neq \{1, 2, 3\}$. Two multisets are equal if they have the same number of each type of element. For multisets, the order of elements still does not matter.

Rules for encoding a selection of n objects from m varieties

Selections	Code words
n = number of items to select	n = number of 0's in code word
m = number of varieties	$m-1$ = number of 1's in code word
Number selected from the first variety	Number of 0's before the first 1
Number selected from the i-th variety, for $1 < i < m$	Number of 0's between the i-1st and i-th 1
Number selected from the last variety	Number of 0's after the last 1

If the mapping of selections to code words is a bijection, then by the bijection rule, the number of distinct code words is equal to the number of distinct selections. If the number of objects to select is n, and the number of varieties of object is m, each code word has n 0's and $m-1$ 1's, for a total of $n+m-1$ bits. The binary string of length $n+m-1$ with exactly $m-1$ 1's is

$$\binom{n+m-1}{m-1}$$

Theorem: Counting Multisets

The number of ways to select n objects from a set of m varieties is

$$\binom{n+m-1}{m-1},$$

if there is no limitation on the number of each variety available and objects of the same variety are indistinguishable.

A set of identical items are called **indistinguishable** because it is impossible to distinguish one of the item from another. A set of different or distinct items are called **distinguishable** because it is possible to distinguish one of the items from the others.

2.10 Assignment problems: Balls in bins

	No restrictions (any positive m and n)	**Max 1 ball per bin** (m must be at least n)	**Same # of balls per bin** (m must evenly divide n)
Indistinguishable	$\binom{n+m-1}{m-1}$	$\binom{m}{n}$	1
Distinguishable	m^n	$P(m,n)$	$\frac{n!}{((n/m)!)^m}$

2.11 Inclusion-exclusion principle

The **principle of inclusion-exclusion** is a technique for determining the cardinality of the union sets that uses the cardinality of each individual set as well as the cardinality of their intersections.

The inclusion-exclusion principle with two sets

Let A and B be two finite sets, then $|A \cup B| = |A| + |B| - |A \cap B|$

The inclusion-exclusion principle with three sets

Let $A, B,$ and C be three finite sets, then

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$$

The inclusion-exclusion principle with an arbitrary number of sets

Let $A_1, A_2, \ldots, A_n$ be a set of n finite sets.

$$\begin{aligned} |A_1 \cup A_2 \cup \cdots \cup A_n| = &\sum_{j=1}^{n} |A_j| \\ &- \sum_{1 \le j \le k \le n} |A_j \cap A_k| \\ &+ \sum_{1 \le j \le j \le l \le n} |A_j \cap A_k \cap A_l| \\ &\vdots \\ &+ (-1)^{n+1} |A_1 \cap A_2 \cap \cdots \cap A_n| \end{aligned}$$

The inclusion-exclusion principle and the sum rule

A collection of sets is **mutually disjoint** if the intersection of every pair of sets in the collection is empty. If we apply the principle of inclusion-exclusion to determine the union of a collection of mutually disjoint sets, then all the terms with the intersections are zero. Thus, for a collection of mutually disjoint sets, the cardinality of the union of the sets is just equal to the sum of the cardinality of each of the individual sets:

$$|A_1 \cup A_2 \cup \cdots \cup A_n| = |A_1| + |A_2| + \cdots + |A_n|.$$

The equation above is a restatement of the sum rule which only applies when the sets are mutually disjoint.

Determining the Cardinality of a Union by Complement

Counting by complement can be used to express the size of the union as:

$$|U| - |\overline{P_1 \cup P_2 \cup \cdots \cup P_n}| = |P_1 \cup P_3 \cup \cdots \cup P_n|$$

3 Advanced Counting

4 Discrete Probability

5 Graphs

6 Trees