离散数学

第一章 基础逻辑符号与推导

命题Proposition(也叫陈述句statement)是一个陈述句(陈述一个事实)，它要么是真的，要么是假的，但不能两者都是

命题变量Proposition Variable是表示命题的字母。常规字母有p、q、r、s、……。示例:r:彼得是男孩

命题逻辑Proposition Logic是处理命题的逻辑领域

逻辑操作符Logic Operators

逻辑符号	Formal Name	Nickname	Symbol	注释/翻译
¬	Negation Operator	NOT	¬	否定运算
∧	Conjunction Operator	AND	∧	逻辑与
∨	Disjunction Operator	OR	∨	逻辑或
⊕	Exclusive-OR Operator	XOR	⊕	异或运算
→	Conditional Statement	Imply	→	条件语句
↔	Biconditional Statement	Equivalent	↔	双条件语句

优先级	逻辑符号
1	¬
2	∧
3	∨ ⊕
4	→
5	↔

Truth tables真值表

列出所有可能的结果，注意从优先级高的依次开始列。

Types of Proposition 命题类型

Tautology 永真式 不论取值永远为真 如 $P \vee \neg P$

Contradiction 矛盾式 不论取值永远为假 如 $P \wedge \neg P$

Contingency 一般命题 真假看取值 如P

Logically Equivalence

等价关系	表达式
Identify Laws	`p ∧ T ≡ p`
等同律	`p ∨ F ≡ p`
Domination Laws	`p ∨ T ≡ T`

等价关系	表达式
支配律	`p ∧ F ≡ F`
Idempotent Laws	`p ∨ p ≡ p`
幂等律	`p ∧ p ≡ p`
Negation Laws	`p ∨ ¬p ≡ T`
否定律	`p ∧ ¬p ≡ F`
Double Negation Law 双重否定律	`¬(¬p) ≡ p`
Commutative Laws	`p ∨ q ≡ q ∨ p`
交换律	`p ∧ q ≡ q ∧ p`
Associative Laws	`p ∨ (q ∨ r) ≡ (p ∨ q) ∨ r`
结合律	`p ∧ (q ∨ r) ≡ (p ∧ q) ∨ r`
Distributive Laws	`p ∨ (q ∧ r) ≡ (p ∨ q) ∧ (p ∨ r)`
分配律	`p ∧ (q ∨ r) ≡ (p ∧ q) ∨ (p ∧ r)`
Absorption Laws	`p ∨ (p ∧ q) ≡ p`
吸收律	`p ∧ (p ∨ q) ≡ p`
De Morgan's Laws	`¬(P1 ∧ P2 ∧ ... ∧ Pn) ≡ ¬P1 ∨ ¬P2 ∨ ... ∨ ¬Pn`
迪摩根律	`¬(P1 ∨ P2 ∨ ... ∨ Pn) ≡ ¬P1 ∧ ¬P2 ∧ ... ∧ ¬Pn`

Some Important Equivalences

等价关系
$P \rightarrow Q \equiv \neg P \vee Q$
$P \rightarrow Q \equiv \neg Q \rightarrow \neg P$
$P \vee Q \equiv \neg P \rightarrow Q$
$P \wedge Q \equiv \neg(P \rightarrow \neg Q)$
$\neg(P \rightarrow Q) \equiv P \wedge \neg Q$
$(P \rightarrow Q) \wedge (P \rightarrow R) \equiv P \rightarrow (Q \wedge R)$
$(P \rightarrow R) \wedge (Q \rightarrow R) \equiv (P \vee Q) \rightarrow R$
$(P \rightarrow Q) \vee (P \rightarrow R) \equiv P \rightarrow (Q \vee R)$
$(P \rightarrow R) \vee (Q \rightarrow R) \equiv (P \wedge Q) \rightarrow R$

第二章 集合

2.1&2.2 Sets and Set Operations

定义：集合是一个无序的对象的组合，集合中的对象被称为集合的成员或元素

有很多种方式表示集合：列出所有元素，设置表达式，韦恩图

常见集合：实数R，自然数N，整数Z，有理数Q。

Set Builder：描述元素作为成员必须具备的属性，如S = {x | P(x)}

韦恩图：矩形表示泛基（全集）U，图形表示集合，点表示元素。

空集指没有元素的集合 $\emptyset$或 { }

只有一个元素的集合被称为单集（Singleton Set）

两个集合当且仅当拥有相同的元素时相当。$\forall x(x \in A \leftrightarrow x \in B)$

常见错误 $\emptyset$ 和{ $\emptyset$}不相等，可以将集合类比为文件夹

A是B的子集（subset）当且仅当A中的每个元素都是B中的元素。记为 $A \subseteq B$

如果$A \subseteq B$ 并且 $B \subseteq A$ 可以推出A=B。

每个非空集合至少有2个子集即集合本身和空集。

当A是B的子集并且A≠B时，A为B的真子集，记为$A \subset B$,逻辑表达式为
$\forall x(x \in A \rightarrow x \in B) \wedge \exists x(x \in B \wedge x \notin A)$

集合内可以有其他集合作为成员。

S的幂集时一个包含S所有子集的集合，记为P（S），一个有n个元素的集合，幂集含有2^n 个元素。

集合S中有n个不同的元素，那么S是一个可数集，n是集合的基数（cardinality），记为|S|。如果一个集合含有无限个元素，为不可数集，如正整数集。

集合是无序的，元组（tuple）是有序的，有序2元组被称为有序对，元组的相等需要每个位置上的元素对应相等。

A×B称为A，B的笛卡尔积（Cartesian Products）$A \times B=\{(a, b) | a \in A \wedge b \in B\}$

A×B并不等于B×A ,除非A，B为空集或A=B。

$A_1 \times A_2 \times \ldots \times A_n=\{(a_1, a_2, \ldots, a_n) | a_i \in A_i, i=1,2, \ldots, n\}$

我们可以利用集合限制量词的定义域，对于一个量词P，和定义域D，定义P的真值集（truth set）为在定义域内的使P（x）为真的元素x构成的集合。记为$\{x \in D | P(x)\}$

$\forall x \in S(P(x))$ 等价于：$\forall x(x \in S \rightarrow P(x))$
$\exists x \in S(P(x))$等价于：$\exists x(x \in S \wedge P(x))$

两个集合可以以多种方式组合，如补(¯)，交(∩)，并(∪)，除(−)，异或($\bigoplus$)

集合运算	数学表示
并集（Union）	$A \cup B=\{x \mid x \in A \vee x \in B\}$
交集（Intersection）	$A \cap B=\{x \mid x \in A \wedge x \in B\}$
差集（Set Difference）	$A-B=\{x \mid x \in A \wedge x \notin B\}$
补集（Complement）	$A^{\prime}=\{x \mid x \notin A\}$
对称差（Symmetric Difference）	$A \oplus B=\{x \mid(x \in A \vee x \in B) \wedge \neg(x \in A \wedge x \in B)\}$

容斥原理：$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{k=1}^{n}(-1)^{k-1}\sum_{1\le i_1<i_2<\ldots<i_k\le n}\left|A_{i_1}\cap A_{i_2}\cap\ldots\cap A_{i_k}\right|$

二元形式：$|A\cup B| = |A| + |B| - |A\cap B|$

集合的运算规则与之前学过的逻辑命题类似

集合运算法则	数学表示
Identity Laws	$A\cup\emptyset = A$
	$A\cap U = A$
Domination Laws	$A\cup U = U$
	$A\cap\emptyset = \emptyset$
Idempotent Laws	$A\cup A = A$
	$A\cap A = A$
Complementation Law	$A\cup A' = U$
	$A\cap A' = \emptyset$
Commutative Laws	$A\cup B = B\cup A$
	$A\cap B = B\cap A$
Associative Laws	$A\cup(B\cup C) = (A\cup B)\cup C$
	$A\cap(B\cap C) = (A\cap B)\cap C$
Distributive Laws	$A\cap(B\cup C) = (A\cap B)\cup(A\cap C)$
	$A\cup(B\cap C) = (A\cup B)\cap(A\cup C)$
De Morgan’s Laws	$(A\cup B)' = A'\cap B'$
	$(A\cap B)' = A'\cup B'$
Absorption Laws	$A\cup(A\cap B) = A$
	$A\cap(A\cup B) = A$
Complement Laws	$A\cup A' = U$
	$A\cap A' = \emptyset$

证明两集合相等有三种方法：

1.真值表

2.证明builder 一样（利用前一章所学对逻辑表达式进行推导）

3.证明$A\subseteq B$并且$B\subseteq A$

2.3 Functions

f是一个函数从集合A映射到集合B，那么A是定义域（domain）,B是值域(codomain)。对于f(a)=b，a为b的前像（preimage），b为a的像(image)。

一个函数由三项元素组成：定义域，值域，映射方向。当且仅当这三个元素全部相同时两函数相等。

严格递增或递减不包含等于的情况。

Floor Function为下取整，Ceiling Function为上取整。

函数运算法则：$(f_1+f_2)(x)=f_1(x)+f_2(x)$ ，$(f_1\cdot f_2)(x)=f_1(x)\cdot f_2(x)$

S的项的表示$f(S)=\{t|\exists s\in S(t=f(s))\}$

一一对应(One-to-One)：如果$a\neq b, f(a)\neq f(b)$　数学表达式为 $\forall a\forall b(f(a)=f(b)\to a=b)$。表示对于一个y只有一个唯一的x与之对应。

满射(onto / surjective)：$\forall y\exists x(f(x)=y)$ 。每一个y都有只要一个x与其对应。

非一一对应函数没有反函数。

复合函数：(f o g)(a) = f(g(a)) 。复合函数不满足交换律，满足结合律。

第五章 关系

5.1&5.2&5.3 关系及其性质&n元关系及其应用&关系表示

关系（relation）定义为:从A到B的二进制关系，有序对，其中:第一个元素来自A，第二个元素来自B。

图（graph）代表A到B的关系，点代表$V\subseteq A\cup B$，边代表$S\subseteq A\times B$

无向图（Undirected Graph）无方向，有向图（Directed Graph）有方向。

一个关系的对应矩阵定义为：

$$m_{ij}=\begin{cases}1 & \text{if } (a_i,b_j)\in R\\ 0 & \text{if } (a_i,b_j)\notin R\end{cases}$$

关系可能具有的六种性质

Symmetric（对称性）：

$$\forall a\forall b\left(((a,b)\in R)\to((b,a)\in R)\right)$$

Asymmetric（非对称性）：

$$\forall a\forall b\left(((a,b)\in R)\to((b,a)\notin R)\right)$$

Antisymmetric（反对称性）：

$$\forall a\forall b\left(((a,b)\in R\land(b,a)\in R)\to(a=b)\right)$$

Reflexive（自反性）：

$$\forall a\left((a,a)\in R\right)$$

Irreflexive（非自反性）：

$$\forall a\left((a\in A)\to((a,a)\notin R)\right)$$

Transitive（传递性）：

$$\forall a\forall b\forall c\left(((a,b)\in R\land(b,c)\in R)\to((a,c)\in R)\right)$$

自反矩阵的主对角线元素全为1，非自反矩阵的主对角线元素全为0。对称性主对角线两端元素一样，非对称性矩阵1沿主对角线对称的全为0，主对角线上全为0.反对称性所有1沿主对角线对称全为0.自反矩阵一定不是非对称。

逆关系（Inverse Relation）：

$$R^{-1}=\{(b,a)|(a,b)\in R\}$$

补集关系（Complementary Relation）：

$$\bar{R}=\{(a,b)|(a,b)\notin R\}$$

关系运算法则

$$(R^{-1})^{-1} = R$$

$$(R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}$$

$$(R_1 \cap R_2)^{-1} = R_1^{-1} \cap R_2^{-1}$$

$$(A \times B)^{-1} = B \times A$$

$$\emptyset^{-1} = \emptyset$$

$$(R)^{-1} = (R^{-1})$$

$$(R_1 - R_2)^{-1} = R_1^{-1} - R_2^{-1}$$

$$\text{If } R_1 \subseteq R_2 \text{ then } R_1^{-1} \subseteq R_2^{-1}$$

如果一个关系中不足以给出传递性的反例，那么该关系就是传递性的。比如只有一个有序对的关系。

两个关系的与或操作为其矩阵对应位分别与或。

两个关系R和S的复合，记作S o R，定义如下：

设：

- R是从集合A到集合B的关系，由(a, b)对表示，其中a ∈ A，b ∈ B。
- S是从集合B到集合C的关系，由(b, c)对表示，其中b ∈ B，c ∈ C。

复合关系S o R包含了有序对(a, c)，其中：

- a ∈ A，c ∈ C，并且存在元素b ∈ B，(a, b) ∈ R，(b, c) ∈ S。

$(x, y) \in S \circ R \Rightarrow \exists z\left((x, z) \in R \wedge (z, y) \in S\right)$

复合关系运算法则

$$(R_1 \circ R_2)^{-1} = R_2^{-1} \circ R_1^{-1}$$

$$F \circ (G \cup H) = (F \circ G) \cup (F \circ H)$$

$$F \circ (G \cap H) \subseteq (F \circ G) \cap (F \circ H)$$

$$(G \cup H) \circ F = (G \circ F) \cup (H \circ F)$$

$$(G \cap H) \circ F \subseteq (G \circ F) \cap (H \circ F)$$

复合关系对应矩阵为其组成部分对应矩阵相乘$M_{R_2 \circ R_1} = M_{R_1} \cdot M_{R_2}$

$$\text{If } R \subseteq S, \text{ then } S \circ R \subseteq S \circ S.$$

$$\text{If } R \subseteq S \text{ and } T \subseteq U, \text{ then } R \circ T \subseteq S \circ U.$$

当且仅当在关系 R 中存在长度为 n 的路径从 x 到 y 时，有序对 (x, y) 属于 R^n。

关系 R 具有传递性当且仅当对于 n > 0，都满足R^n⊆ R。

如果R是可传递的，那么R^n也是可传递的

5.4&5.5 关系的闭包&等价关系

闭包的定义

- 假设 R 是集合 A 上的一个关系。
- 如果关系 S 满足性质 P，则称 S 是相对于性质 P 关于 R 的封闭。
  - S 满足性质 P。
  - S 是包含 R 的、满足性质 P 的每一个关系的子集。

- 向 R 中添加最少的项（有序对）以满足性质 P 的要求。

三种性质的闭包：自反闭包，对称闭包，传递闭包。

r(R)表示R的自反闭包,$r(R) = R \cup D$ D为对角矩阵，即主对角线上全为1的矩阵。

s(R)表示R的对称闭包,$s(R) = R \cup R^{-1}$

t(R)表示R的传递闭包，$t(R) = R^{*}, R^{*} = \sum_{n=1}^{\infty} R^{n} = \sum_{n=1}^{|A|} R^{n}$ （此处累加为交，|A|表示A中的元素个数）

为了简化t(R)的运算，利用Warshall算法。

对于R的矩阵，从第一列开始，标记有1的行，将第一行整行的值加到被标记的行。结束后从第二列开始标记，将第二行加到标记的行。知道最后一列。

示例：

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

第一列二三行有1，将第一行加到二三行上

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

继续以上操作

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

最后的结果就是t（R）。

1. 如果R是自反的，那么s(R) 和t(R)也是自反的。
2. 如果R是对称的，那么t(R) 和r(R)也是对称的。
3. 如果R是传递的，那么r(R) 也是传递的，s(R)不一定是传递的

等价关系：等且仅当关系R为自反，对称，传递时，R为等价关系。

- 相对于关系 R，元素 a 的等价类被表示为$[a]_R$。
- 等价类 $[a]_R$包括所有满足条件 (a, s) ∈ R 的元素 s，即 $[a]_R$ = {s | (a, s) ∈ R}。
- 如果元素 b 属于等价类 $[a]_R$，则称元素 b 是该等价类的代表元素。

  R为等价关系时：aRb,[a]=[b]，[a]∪[b]$\neq \emptyset$ 这三个命题是等价的

集合S_1、S_2、...、S_n 构成A的划分(partition)的条件：

1. 非空：每个子集S_i都是非空的，即S_i≠ ∅。
2. 互不相交：如果i ≠ j，那么子集S_i和S_j是互不相交的，即S_i∩ S_j = ∅。
3. 覆盖A：所有子集Si的并集等于A，即∪S_i = A。

等价关系 R 的等价类将集合 A 划分为互不相交的非空子集合，这些子集的并集是整个集合 A。

- 这个划分通常用 A/R 表示，它被称为"商集"（Quotient set），或者是由 R 引发的 A 的划分，也可以称为"A 模 R"。
- 这个划分是由等价类组成的集合，这些等价类的并集是整个集合 A

对于关系 R，它的自反、对称和传递闭包，表示为 tsr(R) = t(s(r(R)))，是 A 上的一个等价关系。这个等价关系称为由关系 R 引发的等价关系。

5.6 偏序关系

定义：自反，反对称，传递性关系为偏序关系。

R为A上的偏序关系则（A，R）为偏序集（poset）

对于两个元素a,b和一个偏序关系R，如果可以构成aRb或者bRa那么称a和b是可比较的。

当（S，R）是一个偏序集并且任意两个元素都是可比较的时，S被称为全序集(total ordered o)、线性有序集(linear ordered)或简单有序集(simple ordered set)。在这种情况下，（S，R）被称为一个链（chain）。

字典序（Lexicographic Order）先判断最前面的位是否可以构成关系，如果相等则继续判断下一位。

Hasse图：表示一个集合的偏序关系的图。

Hasse图中不存在自反环，因为偏序关系肯定是传递的，所以可以被传递简化的路径也需要被删掉，假设所有的边方向都是向上，所以不用在图中展示边的方向。

绘画Hasse图的标准步骤：

1. 构建poset（A，R）的有向图表示，确保所有弧都指向上方（除非是自环）。
2. 消除所有自环。
3. 从顶部开始逐步消除所有多余的弧，多余的弧是指可以通过其他路径到达的弧，这些路径不包括自环。
4. 消除边的方向，使得所有弧都不再具有方向性，只表示元素之间的关系。

覆盖关系：（S，R）是一个偏序集，存在（x，y），其中 x<y，且没有元素 z 属于 S，使得 x<z<y。（这里的“<”表示严格偏序关系，即满足非自反性）

设（A，R）是一个偏序集，S是A的子集。

$a \in S$，a被称为S的minimal element/ (maximal element)如果不存在$x \in S$，使xRa/aRx。

a被称为S的least element/(greatest element)如果S中的每个元素x，都满足aRx/xRa。

$b \in A$,b被称为lower bound (upper bound) ,如果S中的每个元素x，都满足bRx/bRa。（区别于least element的地方在于不要求属于子集S）

b被称为least upper bound (greatest lower bound)，如果在所有上界/下界组成的集合中为least element/(greatest element)。（同样不一定存在）

一个非空集合的minimal element/ (maximal element)一定存在，least element/(greatest element)不一定存在。

一个链（A,R）被称为良序如果A的每个非空子集都有一个least element。

每一对元素都有最小上界和最大下界的偏序集称为格(Lattice)。

如果 L 是一个格，a 和 b 的最小上界和最大下界可以分别定义为 a∪b 和 a ∩ b。∪和 ∩满足以下性质，其中 a、b 和 c 属于 L。

1. 交换律
   a ∪ b = b ∪ a 和 a ∩ b = b ∩ a
2. 结合律
   a∪(b ∪ c) = (a ∪ b) ∪ c 和 a ∩ (b∩ c) = (a ∩ b) ∩ c
3. 幂等律
   a ∪ a = a 和 a ∩a = a
4. 吸收律
   a ∪ (a ∩ b) = a 和 a ∩ (a∪ b) = a

需要注意的是，∪和 ∩ 不一定需要是“或”和“与”操作符。它们可以是任何满足上述性质的二元运算。

一个格（L，∪，∩）称为有界格（bounded lattice），如果存在元素α、β属于L，对于每个x属于L，满足以下条件：

- x ∪ α = α
- β ∩ x = β

在这里：

- α 被称为L的最大元素，通常用1表示。
- β 被称为L的最小元素，通常用0表示。

如果一个格是有界的，那么以下关系成立：

- 1 是该格的最小上界（lub）。
- 0 是该格的最大下界（glb）。

一个有界格称为互补格（complemented lattice），如果对于每个x属于L，都存在y属于L，满足以下条件：

- x ∪ y = 1
- x ∩ y = 0

在这里：

- y 被称为x的互补（complement），通常用¬x表示。
- 一般情况下，一个元素可能有多个互补。

对偶原则(Principle of dual):当格的偏序关系($\leq$)被替换为($\geq$), ∪ 和 ∩ 被交换时，所有命题真假不变。

至此，前几章所学内容被来连接在一起，从逻辑运算，到集合，到函数，到关系，都可以被放在同意框架下进行理解。

第三章 计数

加法原则，乘法原则

集合容斥原理

排列$P(n,k)=\frac{n!}{(n-k)!}$

组合$C(n,k)=\frac{n!}{k!(n-k)!}$

常用结论

$C(n,r)=C(n,n-r)$

$P(n,r)=C(n,r)*P(r,r)$

$C(n,r)=C(n-1,r)+C(n-1,r-1)$

$\sum_{k=0}^{r}\binom{m}{k}\binom{n}{r-k}=\binom{m+n}{r}$

$\sum_{j=r}^{n}\binom{j}{r}=\binom{n+1}{r+1}$ 证明：在n+1个物品中挑选r+1个物品，可以转化为最后一个选的物品为第r+1个，然后再在前r个里选r个，加上最后一个选的物品为第r+2个，在前r+1个物品里选r个，直到最后一个选的物品为第n+1个，在前n个里选r个。以上式子的每一项求和对应$\sum_{j=0}^{n}\binom{j}{r}$，总的结果等于$\binom{n+1}{r+1}$

计数问题一般算法：

- 首先检查是否可以使用“排列/组合”，否则需要“列出所有可能性”·
- 尝试使用“乘法/加法原理”和“容斥原理”将问题分解成一个小部分

常见题型：

隔板问题例如

从一个装有1美元、2美元、5美元、10美元、20美元、50美元和100美元的现金箱中选出5张钞票有多少种方法?假设:·选择顺序无关·每种面额的钞票无法区分·每种类型的钞票至少5张。

一共有七种类别，每种类别选择的顺序无关，相同种类的选择顺序无关。用六个隔板可以分隔出七个区域。每个区域中放置几个球表示这个类别选择几个。本题有六个隔板，放置五个标记，所以有11种位置，在这11种位置中选取6个隔板或五个标记，最后的答案就是$\binom{11}{6}$ 或$\binom{11}{5}$ 。

总结对于有k类物品，选取r个，当选择顺序无关，相同种类的物品完全相同时可以利用隔板法求解，k-1个隔板，r个标记，结果为$\binom{k+r-1}{r}$或$\binom{k+r-1}{k-1}$

类似问题还有方程x1 + x2 + x3 = 11有多少个解?将x1,x2,x3看成三个种类，放置两个隔板。每个1算一个标记。当要求X1≥1,x2≥2,x3≥3时，在对应的盒子先放入对应的标记，转化为x4 + x5 + x6 = 5。

有些问题只能采取穷举的方式比如：有多少种方法可以把同一本书的6本装进4个相同的盒子里，一个盒子可以装6本书?

本题盒子都是一样的，没法使用隔板法，只能穷举。

生成排列和组合

对于一个有限集合，怎样系统地生成所有组合和排列

对于有n个元素的集合，用长度为n的二进制字符串第k位表示第k个元素是否当前组合或排列中

找寻下一个比特串：在当前比特串的最低位加一

找寻下一个r个元素的组合：

1. i = r
2. while a_i= n - r + i
   2.1 i = i - 1
3. a_i
   = a_i+ 1
4. for j = i + 1 to r
   4.1 a_j= a_i+ j - i

比如对{1,2,3,4,5,6}的4-组合,找寻{1,2,5,6}的下一个组合。

第一步令i=r,定位{1,2,5,6}的最后一位，即6。第二步找寻第一个不等于n-r+i的数。这一步是为了找寻可以增加的数。当a_i=n-r+i时，它代表当前位达到了最大可能的数。比如a_4对应6，a_3对应5，无法再对这些位进行加1操作。当找到第一个不等于的数a_2时，第三步对他进行+1，a_2=3。第四步，对于a_i后的所有位重新赋值为最小可能的值，a_3=4，a_4=5。

生成下一个r个元素的排列：

1. j = n – 1
2. while a_j >a_{j+1}
   2.1 j := j – 1

3. k = n
4. while a_j > a_k
   4.1 k = k – 1
5. interchangea_j and a_k
6. r = n
7. s = j + 1
8. while r > s
   8.1 interchange a_r and a_s
   8.2 r = r – 1 and s = s + 1

比如对362541的下一个排列

第一步令j=n-1,第二步，找到第一个满足a_j <a_{j+1}的a_j ，即找寻第一个升序相邻组，这里为25，a_3=2,a_4=5。第三步令k=n，第四步，找寻第一个满足a_j < a_k ,这里为a_5=4。交换a_j 和 a_k 。前面两步的原因是排列按照字典序排序，要想找寻下一个大的排列，需要将大的数放在前面，对于连续的降序组已是最大字典序的排序，想要找到更大的排列必须要找到升序相邻组。找到a_j 后需要在其之后的降序排列中找到一个比他大的值进行替换，并且要求相邻的下一个排列，应找最小的比a_j大的值。因此第四步找到第一个比a_j大的值，第五步进行交换。六七两步令r=n,s=j+1,第八步循环交换a_r和a_s 第五步交换已经确保了交换后的排列比原先排列字典序大，为了使其是相邻的排列，需要让交换后的排列尽可能小，交换后a_j后为降序排列，通过两两交换，使其变为升序排列，字典序最小。

抽屉原理

如果将N个物体放入k个盒子中，则至少有一个盒子包含至少$\lceil N/k \rceil$个物体

子序列(Subsequence)

假设a_1, a_2，...a_N是实数序列。

该序列的子序列是a_{i_1},a_{i_2},...a_{i_N} 满足1$\leq$ i_1<i_2 <...<i_N $\leq$ N

如果一个序列的每一项都大于它前面的一项，则称为严格递增序列(strictly increasing);如果一个序列的每一项都小于它前面的一项，则称为严格递减序列(strictly decreasing)

定理：每个由n^2+1个不同实数组成的序列都包含一个长度为n + 1的子序列，该子序列要么严格递增，要么严格递减

证明：定义有序对(i_k,d_k),i_k表示从a_k开始的最长递增序列的长度，d_k表示从a_k开始的最长递减序列的长度。

假设没有递增序列和递减序列的长度超过n。一共有n^2+1 个有序对，i_k的取值有n个，d_k的取值有n个，所以不同取值的有序对一共有n^2个。根据抽屉原理，一定有两个有序对值相同。假设为a_m和a_n 对应得有序对。而这俩个值肯定构成递增或递减的关系，所以有序对的值不可能相等。矛盾，所以肯定包含一个长度为n+1的自序列

拉姆齐定理(Ramsey Theory)

六个人中，一定有三个人是共同的朋友或仇人。对于其中一个人来说，根据抽屉原理，剩下的五个人肯定有三个朋友或三个仇人。如果有三个朋友，那么只有这三个人中有一对朋友，那么这六个人中就有三个人是共同的朋友。如果这三个人都是仇人，那么也符合预期。第二种情况同理。

拉姆齐数R(m, n)：在假设聚会上的每对人都是朋友或敌人的情况下，一个聚会上有m个共同朋友或n个共同敌人的最小人数·m和n是大于等于2的正整数

第四章 高级计数技巧

递归关系：序列{an}的递归关系是用一个或多个前元素(a0，...，an−1)表示an的方程。如果序列的项满足递归关系，则称为递归关系的解

常见递归：

哈诺塔：将n层塔从A移动到C需要进行三个步骤1.将第1层到n-1层移到B。2.将第n层从A移到C。3.将B上的n-1层移到C上

所以移动次数A_n=2A_{n-1} +1

斐波那契数列：A_n=A_{n-1}+A_{n-2}

长度为n并且没有m个连续0的位串的个数：长度为n且满足条件的位串可以分为末尾是1，前面有a_{n-1}种位串，末尾是10前面有a_{n-2}种情况。以此类推直到末尾是1加m-1个0，前面是a_{n-m}种位串。递归公式为a_n=a_{n-1}+a_{n-2}+...+a_{n-m}

求解k次常系数线性齐次递归式

k次常系数线性齐次递归式(Linear Homogeneous Recurrence of Degree k with Constant Coefficients)是这种形式的递归式：

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \ldots + c_k a_{n-k} = \sum_{i=1}^{k} c_i a_{n-i}$$

线性指所有的项幂次都是1，齐次(Homogeneous)指没有常数项，次指几项递归式，常系数指c和n无关。

特征根方程：令递归式中的最低次项为x^0 ,每高一次，幂次高一。递归式转化为

$$x^k = c_1 x^{k-1} + c_2 x^{k-2} + \ldots + c_k$$

这个方程被称为特征根方程。

设方程的解为r_1,r_2,...,r_k 当没有重根时$a_n = w_1 r_1^n + w_2 r_2^n + .. w_k r_k^n$ ，各个w的值根据前k个a_n的值解方程得出。

当有重根时,设有m个重根r_c ,则这m个重根对应的解为$(c_1 + c_2 n + \ldots c_m n^{k-1}) r_c^n$ 。

求解非齐次线性递归

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \ldots + c_k a_{n-k} + F(n)$$

如果$a_n^{(p)}$是一个特解，那么方程的通解为$a_n^{(k)} + a_n^{(h)}$，其中$a_n^{(h)}$是关联齐次递归式$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \ldots + c_k a_{n-k}$的解

当F(n)为如下形式时

$$F(n) = (b_t n^t + b_{t-1} n^{t-1} + \ldots + b_1 n + b_0) s^n$$

当s不是关联齐次递归式的根时，特解具有以下形式

$$(p_t n^t + p_{t-1} n^{t-1} + \ldots + p_1 n + p_0) s^n$$

当s是关联齐次递归式的m重根时，特解具有以下形式

$$n^m (p_t n^t + p_{t-1} n^{t-1} + \ldots + p_1 n + p_0) s^n$$

母函数

母函数(G(x))通过将序列的项编码为变量x在形式幂级数中的幂系数来有效地表示序列

级数$G(x) = \sum_{k=0}^{n} a_k x^k$,x^k的系数a_k对应序列中的值。

如{1，1，1，1...}对应$\sum_{k=0}^{\infty} x^k$ =$\frac{1}{1-x}$

对于$f(x) = \sum_{k=0}^{\infty} a_k x^k$,$g(x) = \sum_{k=0}^{\infty} b_k x^k$ 有

$f(x) + g(x) = \sum_{k=0}^{\infty} (a_k + b_k) x^k$

$f(x) g(x) = \sum_{k=0}^{\infty} \sum_{j=0}^{k} a_j b_{k-j} x^k$

例：$h(x) = \frac{1}{(1-x)^2}$,给出 $h(x) = \sum_{k=0}^{\infty} a_k x^k$ 的系数

$$\frac{1}{(1-x)^2}=\frac{1}{1-x}*\frac{1}{1-x}$$

$$h(x)=f(x)g(x)$$

$$f(x)g(x)=\sum_{k=0}^{\infty}\sum_{j=0}^{k}x^k=\sum_{k=0}^{\infty}(k+1)x^k$$

因此a_k=k+1。

母函数运用：$e_1+e_2+e_3=17$,e_1,e_2,e_3为非负整数，并且满足$2\le e_1\le 5,3\le e_2\le 6,4\le e_3\le 7$

通过母函数$\sum_{k=0}^{\infty}a_kx^k=(x^2+x^3+x^4+x^5)(x^3+x^4+x^5+x^6)(x^4+x^5+x^6+x^7)$

x^{17}的系数就是答案。

利用母函数求递归式通项公式

设递归式的母函数为G(x)=$\sum_{k=0}^{\infty}a_kx^k$,对于k次递归式，单独取出$a_0$到$a_{k-1}x^{k-1}$ 再配出 $\sum_{k=0}^{\infty}a_kx^k$。最后解出G(x)的表达式，将G(x)展开为级数，即可得出递归式的通项公式。

例：$a_n=-2a_{n-1}-a_{n-2},a_0=5,a_1=-6$

$$G(x)=\sum_{k=0}^{\infty}a_kx^k$$

$$G(x)=a_0x^0+a_1x^1+\sum_{k=2}^{\infty}a_kx^k$$

这一步就是根据递归式的次数拆出对应的项。这里的二次递归式，所以拆出前两个项

$$G(x)=a_0+a_1x+\sum_{k=2}^{\infty}(-2a_{k-1}-a_{k-2})x^k$$

$$G(x)=a_0+a_1x-2\sum_{k=2}^{\infty}a_{k-1}x^k-\sum_{k=2}^{\infty}a_{k-2}x^k$$

$$G(x)=a_0+a_1x-2\sum_{k=2}^{\infty}a_{k-1}x^k-\sum_{k=2}^{\infty}a_{k-2}x^k$$

$$G(x)=a_0+a_1x+2a_0x-2a_0x-2x\sum_{k=1}^{\infty}a_{k-1}x^{k-1}-x^2\sum_{k=2}^{\infty}a_{k-2}x^{k-2}$$

$$G(x)=a_0+a_1x+2a_0x-2x\sum_{k=1}^{\infty}a_{k-1}x^{k-1}-x^2\sum_{k=2}^{\infty}a_{k-2}x^{k-2}$$

配凑出新的G(x)来解出G(x)

$$G(x)=5+4x-2xG(x)-x^2G(x)$$

$$G(x)=\frac{5+4x}{(x^2+2x+1)}$$

$$G(x)=\frac{5+4x}{(1+x)^2}$$

得出G(x)的解析式后需要还原至级数形式，需要利用以下公式

$$\sum_{k=0}^{\infty}a_kx^k=\frac{1}{1-ax}$$

$$f(x)g(x)=\sum_{k=0}^{\infty}\sum_{j=0}^{k}a_jb_{k-j}x^k$$

先拆分为符合利用公式的形式

$$G(x)=\frac{5+4x}{(1+x)^2}$$

$$G(x)=\frac{5(1+x)-x}{(1+x)^2}$$

$$G(x)=\frac{5}{(1+x)}-\frac{x}{(1+x)^2}$$

$$G(x)=\sum_{k=0}^{\infty}5(-1)^kx^k-\frac{x}{(1+x)^2}$$

$$G(x)=\sum_{k=0}^{\infty}5(-1)^kx^k+\sum_{k=0}^{\infty}k(-1)^kx^k$$

$$G(x)=\sum_{k=0}^{\infty}\left(5(-1)^k+k(-1)^k\right)x^k$$

$$a^n=5(-1)^n+n(-1)^n$$

总结：利用母函数求递归式通项主要分为

- 1.写出母函数，根据递归式的次数拆出前几项
- 2.配凑还原母函数，解出母函数
- 3.将母函数级数展开得到通项公式。（主要用到之前提到的两个公式）

第九章 图

图的性质

图的属性:点(vertices),边(edge)

图分为有向图(Directed Graph)，无向图(Undirected Graph);简单图(Simple Graph)两个点之间最多一条边，复合图(Multigraph)两个点之间可以有多条边。有环图(loop)，无环图。注意：这里的loop指自环，即自己指向自己的边，和下面的cycle区分。

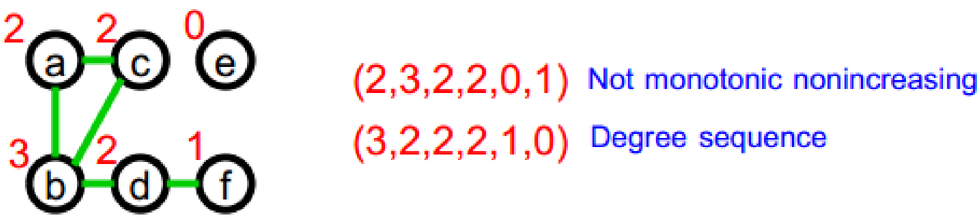
Undirected		No Loop	Loop
	Single Edge	Simple Graph	/
	Multiple Edge	Multigraph	Pesudograph (Multigraph)

Directed		No Loop	Loop
	Single Edge	Simple Directed Graph	/
	Multiple Edge	Directed Multigraph	Mixed Graph

无向图的相邻(adjacent)只要求有一条路径相连，有向图的相邻要求可以通过路径到达(有方向性)。

度(degree)在无向图中为包含该顶点的边数(相邻顶点数)，有向图中分为出度和入度。

度序列(degree sequence)是无向图中顶点度的单调不递增序列。



在任何图中，所有度之和等于边数的两倍

无向图具有偶数个度为奇数的点

对于任何一个有向图，出度之和等于入度之和等于总边数。

如果一个路径上的所有点都不相同，那么这是一个简单(simple)路径

环是将一个点连接到自身的路径。如果环除了起点和终点其他点都没有重复，那么是一个简单环。

如果一个图没有环，那么称为无环图(acyclic),一个没有环的有向图被称为有向无环图(DAG)

当有一条路径可以连接两个顶点时，称为这两个顶点是连接(connected)的。注意：区分于相邻

当无向图中对于每一个顶点都存在至少一条路径可以连接到其他所有顶点，那么这是一个连通图。

当有向图考虑方向并且是连通时称为强连通，不考虑方向并且连通时称为弱连通。

对于某个性质，没有比起更小/更大的图符合该性质，那么这个图被称为满足某性质的最小/最大图

对图的操作（本章重点）

$$\overline{G1} = (V1, \{uv | u \neq v, uv \notin E1\})$$

$$G1 \cap G2 = (V1 \cap V2, E1 \cap E2)$$

$$G1 \cup G2 = (V1 \cup V2, E1 \cup E2)$$

$$G1 + G2 = (V1 \cup V2, E1 \cup E2 \cup \{uv | u \in V1 \text{ and } v \in V2\})$$

$$G1 \times G2 = (V1 \times V2, \{((u1, u2), (v1, v2)) | (u1 = v1 \text{ and } \{u2, v2\} \in E2) \text{ or } (u2 = v2 \text{ and } \{u1, v1\} \in E1)\})$$

完全图（Complete Graph）记作K_n，任意两个顶点之间都有边。

环图(Cycle Graph)记作C_n ,所有顶点构成一个环

轮图(Wheel Graph)记作W_n ,在环图的基础上增加一个和所有顶点相邻的中心。

立方(cube)记作Q_n， n维空间中的完全方体。Q_0为一个点。

二分图(Bipartite Graph) ,如果一个图的所有点都可以分为两部分并且任何两个相邻的顶点都属于不同的部分，那么此图为二分图。一个点的图肯定为二分图。

判断一个图是否是二分图，先找图中是否存在K_3子图。如果存在则一定不是。

完全二分图(Complete Bipartite Graph) ,当一个二分图被分为分别有m，n个顶点的部分并且任意一个点都和另一部分的所有点相邻时，称为完全二分图$K_{n,m}$

树是一个无向，连通，无环的图。n个顶点的树具有n-1条边。

森林是无向，不连通，无环的图。

如果一个树具有至少两个顶点，那么至少有两个叶节点。

设G是一个有n个顶点的图。那么以下是等价的:

- G是树。
- G是一个极大无环图

- G是最小连通图
- G是无环的，它有n - 1条边
- G是连通的，它有n−1条边
- 在G的任意两个不同的顶点之间存在一条唯一的路径

连通图G中的生成树是G的子图H，它包含了G的所有顶点，也是一棵树

最小生成树是总路径消耗最少的生成树

同构图(Isomorphism) 可以在两个图的顶点和边之间建立一一映射则这两个图为同构图。

欧拉通路

只经过每条边一次的路径为欧拉通路

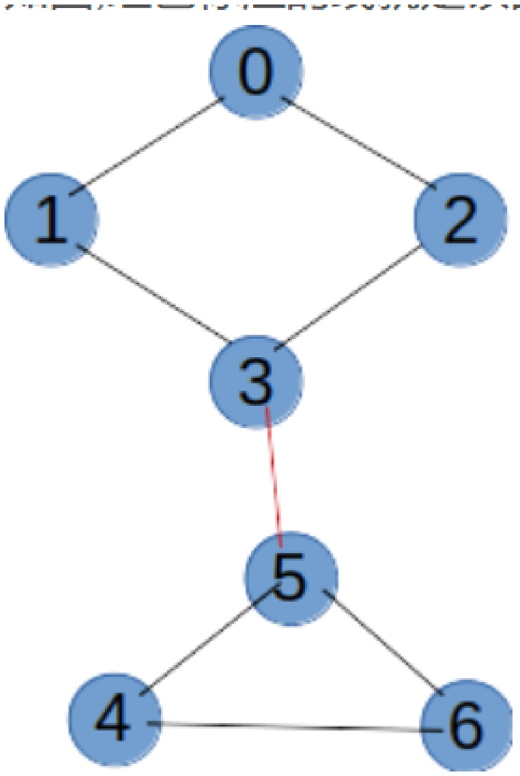
欧拉通路的中间节点度都为偶数，当欧拉通路为环时，起始点度为偶数，否则起始点和结束点度为奇数。

至少有两个顶点的连通图有欧拉环路当且仅当它的每个顶点的度都为偶数。

一个连通图有欧拉路径但没有欧拉环路当且仅当它恰好有两个顶点度为奇数。

找寻欧拉通路利用Fleury’s Algorithm

- 1.如果没有奇数度点，任选一个地方开始。如果有两个奇数度点，选择其中一个作为起点，另一个为终点。
- 2.选择一条非桥的边（在图论中，桥的定义为如果去掉这条边会使整个图的联通分量发生变化，那么这条边就是桥。联通分量指整个图由几个联通图构成，比如下图(3,5)就是桥，去掉他会使整个图从一个联通图变为两个联通图）这一步是为了防止选择桥后无法返回



- 3.当所有边都被遍历时停止，完成欧拉通路。

找寻欧拉环路通过Hierholzer’s Algorithm

- 1.选择节点v作为起始节点
- 2.用未访问过的边形成一个环路，以v结束(去掉访问边)
- 3.当所有的边都被遍历时，停止
  - a)在之前的循环中找到一个与未访问过的边连接的节点u为起点，
  - b)使用未访问的边形成一个环路，并以u结束(删除已访问的边)

- c)在节点u处合并两个环路

哈密顿通路

如果一条路径可以访问所有顶点有且仅有一次，那么这条路径被称为哈密顿通路。

Dirac’s Theorem：如果一个有n($n \geq 3$)个顶点的简单图的每个点的度都≥ $\frac{n}{2}$,那么存在哈密顿环路

Ore’s Theorem：如果一个有n($n \geq 3$)个顶点的简单图的每一对非邻接点的度之和都≥ n，那么存在哈密顿环路

以上两条都是充分条件而非必要条件，没有好的算法来寻找哈密顿通路。

Planar Graph

Planar Graph是指可以在平面上画出没有边相交的图形，在平面上画的没有边相交的Planar Graph称为Plane Graph

Planar Graph图可以有多种绘画形式，并不是所有形式都是Plane Graph。

在平面上绘制的图形也被称为嵌入（embedded）在平面上

一个Plane Graph会把平面分割为几个区域（包含外部无界区域）

图G中与区域R相关的顶点和边构成了G的一个子图，称为R的边界(boundary)

环路属于两个区域的边界，桥只属于外部无界区域。

欧拉定理：如果G是一个连通的平面简单图，有e条边，v个顶点，r个区域，则有r=e-v+2

欧拉定理的扩展：

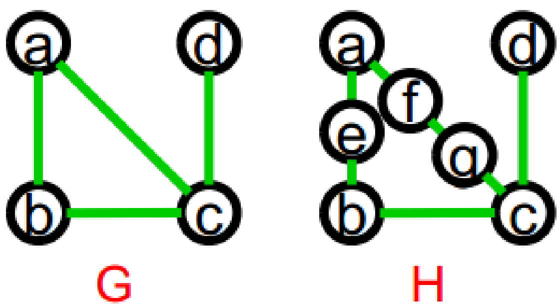
- 一个连通的平面简单图，必有一个顶点的度数不超过5。
- 如果连通的平面简单图有e条边和v个顶点，且v≥3，则e≤3v - 6
- 若连通的平面简单图有e条边和v个顶点，且v≥3，且没有长度为3的回路，则e≤2v - 4

这些扩展可以用来判断一个图是否是平面图

同胚(Homeomorphic)

如果一个图是平面的，则该图是通过去掉一条边{u,v}并添加一个边为{u,w}和{w,v}的新顶点w得到的任意图和其是同胚的。

即去掉一条边的内部点不影响和原图的同胚性



Kuratowski’s Theorem：如果一个图包含一个同胚于$K_{3,3}$或K_5的子图，那么这个图就是非平面的

着色

着色问题：相邻的区域不可以用相同的颜色着色。

用点代表地图区域，边代表两区域相邻。

Chromatic number (χ(G)):使图G完成着色的最小颜色数。

没有色数的表达公式。

这块的考点主要为将问题转化为着色问题

如：一个航班需要机场的一个登机口。该航班需要多少个登机口。

可以将航班转化为顶点，在同一个时间需要登机口转化为边。问题即变成了图的着色问题。