**数理逻辑**

**命题逻辑基本概念**

# 判断句子是否是命题

**命题**：非真即假的陈述句

**真值**：真或假；任何命题的真值都是唯一的

简单/原子命题：不能被分解为更简单的命题

**复合命题**：简单命题+联结词

“我正在说假话”不是命题（既不能为真也不能为假，悖论）

“火星上有生物”是命题

# 命题符号化

半形式化+连结词->形式化

**否定联结词**

**合取联结词**：虽然…但是…，一面…一面…

**析取联结词**：自然语言“或”的二义性（相容或/排斥或）

**蕴涵联结词**：前件、后件，前真后假为假

p->q：除非q才p；除非q，否则非p；只有q，才p；p仅当q（p发生则q一定发生）

“只有a能被4整除，a才能被2整除“真值与a的取值有关，不能简单认为是错的

**等价联结词**：同真同假为真

联结词的优先级：一元联结词（非）优先级高

# 求命题的真值与公式的赋值

命题常项/命题常元：真值确定（一般意义上的命题）

命题变项/命题变元：真值可以变化（不是命题）

合式公式：命题变/常项+联结词

公式的层次：单个命题变项0层，联结词操作一次加1层

赋值/解释：成真赋值，成假赋值

# 判断公式的类型

真值表；重言式/永真式；矛盾式/永假式；可满足式（不是矛盾式）；哑元

**命题逻辑基本概念**

# 等值式

等值式：重言等价式

不是联结词

等值式模式：双重否定律；幂等律；交换律；结合律；分配律；德摩根律；吸收律；零律；同一律；排中律；矛盾律；蕴涵等值式；等价等值式；假言易位；等价否定等值式；归谬论

置换规则

证明等值式不成立：找一个赋值使两公式真值不同

赋值时应将哑元也考虑在内

# 析取范式和合取范式

文字（变量及其否定）；简单析取式；简单合取式

单个文字即是简单合取式也是简单析取式

析取范式：有限个简单合取式的析取->矛盾式

合取范式：有限个简单析取式的合取->重言式

一个命题公式的合取范式和析取范式不唯一

极小项：简单合取式mi， i为成真赋值

极大项：简单析取式Mi，i为成假赋值

n个命题变项可以产生2n个极大项/极小项

主析取范式：极小项的析取

主合取范式：极大项的合取

求主析取范式/主合取范式：利用同一律补加缺失的命题变项（极大项/极小项要包含命题里的所有文字）

主析取范式->成真赋值

主合取范式->成假赋值

**命题逻辑的推理理论**

# 推理的形式结构

推理：前提->结论

推理正确/有效：蕴涵式是重言式，得到有效的结论

推理定律：附加律；化简律；假言推理；拒取式；析取三段论；假言三段论；等价三段论；构造性二难（一般形式，特殊形式）；破坏性二难

# 自然推理系统P

形式系统=形式语言系统+形式推理系统

两类形式系统：自然推理系统，公理推理系统

自然推理系统P：字母表，合式公式，推理规则（前提引入、结论引入、置换规则、合取规则+推理定律）；得到的结论都是有效的，前提为真则结论为真，前提为假时结论不一定为真

注意证明推理过程的表示

附加前提证明法（附加前提引入），反证法/归谬法（结论的否定引入）

**一阶逻辑基本概念**

一阶逻辑命题符号化的3个基本要素：个体词、谓词、量词

个体词：个体常项（a,b,c…）、个体变项（x,y,z…）；个体域，全总个体域

谓词：谓词常项、谓词变项；0元谓词：不带个体变项（x,y,z…），0元谓词常项=命题

量词：全称量词；存在量词

特性谓词：限制个体域

默认采用全总个体域，一般要先限制个体域

**计算模型**

**语言和文法**

# 引言

形式语言

文法：确定/产生形式语言的规则

# 短语结构文法

字母表Σ，符号，词，空串λ，所有词的集合Σ\*，语言（Σ\*的子集）

短语结构文法：G=(Σ, T, S, P)

T：终结符集，S起始符，P产生式集合，N=Σ-T非终结符集，产生式左端至少包含一个非终结符

派生：可直接派生，可派生，派生（得到的过程序列）

G生成的语言L(G)：从S出发能派生出的所有终结符串的集合

给定文法构造语言：简单

给定语言构造文法（做题）

# 短语结构文法类型

0型文法：对产生式没有限制

1型文法：上下文有关

除了，其他产生式都是非缔约的（串长度不减少）

正则文法G，只有G的产生式有时，由G生成的语言中才有

所有产生式都是非缔约的，则为1型文法（另一种定义）

非缔约文法是1型文法，只是不能生成

2型文法：上下文无关

3型文法：正则文法

文法->语言

# 派生树（语法分析树）

表示上下文无关文法生成的语言的派生过程

# 巴科斯-诺尔范式（BNF）

<A>::=<A>a|a|<A><B>，一次表示多个产生式

**带输出的有限状态机**

# 引言

状态表：状态，状态+输入->下一状态+输出

# 带输出的有限状态机

有限状态机*M = (S, I, O, f, g, s0)*

*S*: 有限状态集；*I*: 有限输入字母表；*O*: 有限输出字母表；*f : SI S；g : SI O；*

*s0：*初始状态

输入可以看作一个字符串，连续进行状态转移，*g*输出一个字符串

状态机识别语言L：L中的字符串输入进M，M的最后一位输出为1

# 不带输出的有限状态机

不带输出，有终结状态

输入串，能达到终结状态则可识别该串

确定性/非确定性

# 串的集合

连接(AB)：A中的串和B中的串组合（A串在前B串在后）；An

克林闭包：，A中任意多个串的连接组成的集合

# 确定性有限状态自动机（不带输出的有限状态机）

有限状态自动机*M = (S, I, f, s0, F)，F*：终结状态集

可以连续进行状态转移

# 确定性有限状态自动机识别语言

输入一个串，能转移到终结状态，可识别的所有串的集合L(M)；识别相同语言则两自动机等价

# 非确定性有限状态自动机识别语言

输入+状态可以转移到多个状态

输入字符串，连续状态转移后有多条状态转移路径，只要有一个最终到达了终结状态，就可识别该字符串

非确定性有限状态自动机能识别的语言也可以被确定性有限状态自动机识别

由非确定性有限状态自动机得到确定性有限状态自动机

# 语言的识别

# 引言

正则集合：能被有限状态自动机识别的集合；以任意顺序通过对空集，空串和单字符串的连接、并或克林闭包构造出来

正则表达式：空集、空串是正则表达式；x在I中，x是正则表达式；正则表达式的连接、并和克林闭包是正则表达式

正则表达式表达正则集合，二者差不多可以混用

求集合的正则表达式；

# 克林定理

一个集合是正则集合等价于其可以被一个有限状态自动机识别

识别A，B的有限状态自动机已知

1. 构造识别的有限状态自动机
2. 构造识别{λ}的有限状态自动机
3. 构造识别{a}的有限状态自动机
4. 构造识别AB的有限状态自动机

A终结状态前状态连B起始状态；

如果A为终止状态（自动机可识别空串）：A起始状态连B起始状态后状态

1. 构造识别AB的有限状态自动机

选一个自动机

1. 构造识别A\*的有限状态自动机

要识别空串，读完一段要能够循环读下一段

# 正则集合和正则文法

正则文法⬄正则集合（语言）⬄有限状态自动机

根据文法构造自动机；根据自动机构造文法：产生式⬄状态转移

{0n1n}非正则（有状态转移环路）不能被“有限”状态自动机识别

# 图灵机

# 图灵机的定义

图灵机*T=(S, I, f, s0)*

*I*：字母表，包括空白符B

*f :* 5元组(s, x, s’, x’, d)，转移规则

# 用图灵机识别集合

终结状态：不是任何五元组的第一个状态

把字符串写在纸带上，让图灵机从初始位置（最左边非空白符）开始运行，如果能在终结状态停机，这说明图灵机能识别字符串

根据识别的语言构造图灵机（可以添加不在字符表中的标记）：正则集合⬄有限状态自动机⬄图灵机（始终向右移动）

图灵机可以识别非正则集合

# 用图灵机计算函数

整数的一元表示：0=>1, 1=>11, 2=>111…

对任何多带图灵机，存在一个单带图灵机，二者完全等价

# 不同类型的图灵机

非确定性/确定性…

# 丘奇-图灵论题

对于任何可用有效算法来求解的问题，都存在解该问题的图灵机

# 计算复杂性、可计算性和可判定性

判定问题：判断某个特定类型的命题是否为真

可判定性（可解/不可解）：是否存在有效的算法能够判断判定问题的某个解是否正确

停机问题（不可解）：对于给定的输入字符串x，图灵机T最后是否能停机

可计算性（可计算/不可计算）：可计算：函数能够被图灵机计算

一个判定问题是可解的⬄构造的函数是可计算的

P类问题：判定问题能由确定性的图灵机在多项式时间内求解（多项式时间问题）

NP类问题：判定问题能由非确定性的图灵机在多项式时间内求解（非确定性的多项式时间问题）

NP完全问题