§1.2 命题公式及其赋值

例 已知命题 p：北京比天津人多

q：$2+2=4$

据此，求命题 $(p\rightarrow q)\wedge p\wedge\neg q$ 的真值。

解 首先，命题p,q的真值分别为1, 1。把真值形式地带入上述命题

$(1\rightarrow 1)\wedge 1\wedge\neg 1$，由此得真值为0。 ▌

不论p,q是什么样的命题，由表达式 $(p\rightarrow q)\wedge p\wedge\neg q$ 确定的命题，其真值永远为假。

这个结果就是反证法(归谬法)的理论依据

命题常项（命题常元）：简单命题。

命题变项（命题变元）：真值可变化的陈述句。

命题变项的表示符号与命题常项相同，可视为取值为0或1的变量，命题常项相当于常数变量。

前面提到的表达式

$$(p\to q)\wedge p\wedge\neg q$$

可视为以 p,q 为自变量的二元函数，p,q 的取值范围是任意命题。p,q 的取值一旦确定，则表达式变为命题。故该函数的函数值为真值。

定义 将命题变相用联结词和圆括号按照下属规则联结而成的符号串称为**合式公式**(命题公式)：

（1）单个命题变项或命题常项是合式公式，称为原子命题公式；

（2）若 A 是合式公式，则 $\neg A$ 也是合式公式；

（3）若 A, B 是合式公式，则 $A\vee B, A\wedge B, A\rightarrow B, A\leftrightarrow B$ 也是合式公式；

（4）只有有限次地应用步骤(1)~(3)得到的符号串才是合式公式。

注 （1）一般用$p, q, r, \cdots$表示命题常项和命题变项，它们通过联结词和圆括号组成合式公式，而命题常项和命题变项是我们的研究对象，故称$p, q, r, \cdots$、联结词以及圆括号为**对象语言符号**。

（2）一般用$A, B, C, \cdots$表示合式公式，合式公式由对象语言构成，故称$A, B, C, \cdots$为**元语言符号**。

（3）完全由符号构成，为了特定应用而人为设计的语言称为**形式语言**。

为了方便研究合式公式，需要确定联结词以及合式公式的运算等级，故有

定义 合式公式(命题公式)的**层数**按下述规则确定：

（1）单命题变项或命题常项是 0 层公式；

（2）若 B 是 n 层合式公式，则$\neg B$ 是 n+1层公式；

（3）若 B, C 分别为 i 层, j 层合式公式，则 $B\vee C$, $B\wedge C$, $B\rightarrow C$, $B\leftrightarrow C$ 是 n+1 层公式，这里$n = \max\{i, j\}$。

例 $(\neg p\wedge q)\rightarrow r$，$(\neg(p\rightarrow\neg q))\wedge((r\vee s)\leftrightarrow \neg p)$

3 层　　　　　　4 层

一般地，命题公式没有真值，故不是命题。但指定公式中的命题变项的真值后，公式就有了确定的真值。这样，公式变为命题。

定义 设$p_1, p_2, \cdots, p_n$是出现在命题公式A中的全部命题变项。给$p_1, p_2, \cdots, p_n$各指定一个真值，称为对A的一个**赋值**或**解释**。使A为**真**（**假**）的赋值称为**成真**（**假**）**赋值**。

注 设$\alpha_1\alpha_2\cdots\alpha_n$是一个赋值，则约定

$$p_1=\alpha_1, p_2=\alpha_2, \cdots, p_n=\alpha_n \text{ 或 } p=\alpha_1, q=\alpha_2, r=\alpha_3, \cdots$$

真值表：罗列命题公式A在所有赋值下取值的表。 按照二进制数从小到大排列赋值，从低到高排列各个层次，直到整个公式。

例 $(p\wedge\neg p)\leftrightarrow(q\wedge\neg q)$

p	q	$\neg p$	$\neg q$	$p\wedge\neg p$	$q\wedge\neg q$	$(p\wedge\neg p)\leftrightarrow(q\wedge\neg q)$
0	0	1	1	2	2	3
0	0	1	1	0	0	1
0	1	1	0	0	0	1
1	0	0	1	0	0	1
1	1	0	0	0	0	1

例 $(\neg p \wedge q) \to \neg r$

p	q	r	$\neg p$	$\neg r$	$\neg p \wedge q$	$(\neg p \wedge q) \to \neg r$
0	0	0	1	1	2	3
0	0	0	1	1	0	1
0	0	1	1	0	0	1
0	1	0	1	1	1	1
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	1	0	0	0	1
1	1	0	0	1	0	1
1	1	1	0	0	0	1

例 $\neg(p\rightarrow q)\wedge q\wedge r$

p	q	r	$p\rightarrow q$	$\neg(p\rightarrow q)$	$\neg(p\rightarrow q)\wedge q$	$\neg(p\rightarrow q)\wedge q\wedge r$
0	0	0	1	1	2	3
0	0	0	1	0	0	0
0	0	1	1	0	0	0
0	1	0	1	0	0	0
0	1	1	1	0	0	0
1	0	0	0	1	0	0
1	0	1	0	1	0	0
1	1	0	1	0	0	0
1	1	1	1	0	0	0

注（1）上述第三例的公式中，1层与2层的子公式不含命题变项r，因此r的取值不影响这几个子公式。可以称r是这几个子公式的**哑元**。

（2）包含n个命题变项的公式有2^n个赋值。

定义 在各种赋值下取值均为真的命题公 式称为**重言式**或**永真式**，在各种赋值下取值均为假的命题公式称为**矛盾式**或**永假式**，存在成真赋值的命题公式称为**可满足式**。

例 判断公式的类型 $(p\to q)\leftrightarrow(\neg q\to\neg p)$ 。

威廉·莱布尼茨

（Wilhelm Leibniz）

1646—1716

德国哲学家、数学家，被誉为十七世纪的亚里士多德。一般公认为是数理逻辑的缔造者。同时，他和牛顿先后独立发现了微积分，而且他所使用的微积分的数学符号被更广泛的使用。莱布尼茨还发明并完善了二进制。

莱布尼茨在政治学、法学、伦理学、神学、哲学、历史学、语言学诸多方向都留下了著作。

小结：

1. 熟练掌握将命题符号化的方法

   命题的真值，联结词

2. 熟练掌握命题公式类型的判断方法

   命题公式、赋值、重言式与类型判断